■ Le donne nel mondo Linux sono tante ■ Musica con la Raspberry Pi

LINUX PRO

Diventa protagonista della programmazione Open Source

Assalto alla privacy
Ecco chi cerca di fregarci i dati e chi lotta per difenderci **p.16**

ACCADEMIA DEL CODICE

8 pagine di pura programmazione

LinuxPro159-Novembre2015-Mensile-**5.90€**


LA DISTRO PERFETTA!

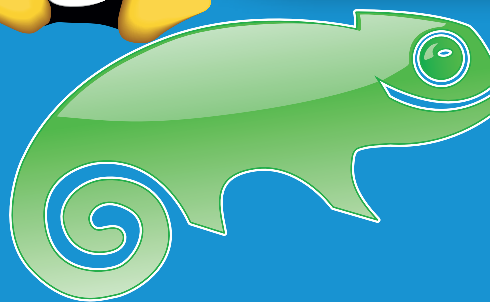
Entra nel mondo Linux con la versione ideale che ti aiutiamo a scegliere noi!

DVD ROM 8GB



INSTALLA ORA LA DISTRO DELL'ANNO!





I contenuti del DVD

Programmi
- Cube 2
- Dust Racing 2D
- Kid3
- LibreOffice 5.0.2
- Lollypop
- Sauerbraten
- Scribus
- Tanks of Freedom
- Tmux

Distribuzioni
- Mageia 5.0
- NethServer 6.6
- Ubuntu Server 14.04.3 LTS
- ClearOS 7.1
- Zentyal 4.1



TARIFFA R.O.C. - POSTE ITALIANE SPA - ABB. POST. - D.L. 353/2003 (CONV. IN L. 27/02/2004 N. 46) ART. 1, COMMA 1, S/NA - DISTRIBUTORE: PRESS-DI, MILANO

Una distribuzione fa la differenza

Che siate un lettore che ha preso in mano la rivista per la prima volta (grazie, promettiamo che non vi deluderemo!) o che siate fedeli lettori fin dai primi numeri (grazie, promettiamo di darvi sempre di più!), la scelta della distribuzione Linux da usare è decisiva. Ognuna ha qualità specifiche, pregi e difetti e ha funzioni a cui assolutamente non si può rinunciare, quindi selezionare il meglio è un punto di partenza essenziale per usare Linux. Per questo motivo, come ogni anno abbiamo passato in rassegna le migliori distribuzioni degli ultimi 12 mesi e le abbiamo analizzate per capire cosa offrono a quale tipo di utente. Ce n'è davvero per tutti i gusti... Gli utenti "mainstream" possono scegliere fra le distro più note grazie agli approfondimenti dei nostri esperti. Gli utenti che vogliono invece mettere le mani sulle migliorie non appena sono disponibili possono lustrarsi gli occhi con le migliori rolling release in circolazione. Chi invece deve gestire server professionali e per i servizi Internet può farsi un'idea di cosa è meglio per il proprio business e per i propri clienti con l'analisi delle più interessanti distro per server. Come ogni anno, le nostre scelte stimolano la discussione su quale distribuzione è davvero ideale e la nostra posizione è sempre la stessa: a ognuno la sua! Anzitutto, ognuno ha i propri gusti e le proprie esigenze, quindi quello che va bene per la maggior parte degli utenti potrebbe non andare bene per altri. Inoltre nessuno ci obbliga a installare una sola distribuzione: grazie alla flessibilità di Linux perché non provarne diverse e sfruttare il meglio da ognuna?

La redazione di Linux Pro





CONTATTI

Domande alla redazione: **redazione@linuxpro.it**
Abbonamenti: **abbonamenti@linuxpro.it**
Arretrati: **arretrati@linuxpro.it**
Problemi con il DVD: **aiutocd@sprea.it**
Sito Web: **www.linuxpro.it**
Oppure inviate le vostre lettere a:
***Linux Pro*, Sprea S.p.A.,**
Via Torino 51, 20063 Cernusco S/N
Telefono: 02.92432.1

LINUX PRO 159

Sommario LINUX PRO

Benvenuti nel centocinquantanovesimo numero di Linux Pro, la guida definitiva a Linux e al mondo Open Source

In primo piano







Sommario

IL PROSSIMO NUMERO ESCE L'11 DICEMBRE

IL DVD IN BREVE

LATO A

DISTRIBUZIONI
- Mageia 5.0



LATO B

DISTRIBUZIONI
- ClearOS 7.1
- NethServer 6.6
- Ubuntu Server 14.04.3 LTS
- Zentyal 4.1

RIVISTA
- Cube 2
- CVAssistant 2.0
- Dust Racing 1.11
- Kid3 3.3
- LibreOffice 5.0.2
- Lollypop 0.9.52
- qBittorrent .2.3
- Qt5CT 0.19
- Scribus 1.5
- Tank of Freedom
- TheSSS 13.1
- Tmux 2.0
- uGet 2.0.2

Quando trovi questo bollo negli articoli, cerca il software nel DVD



Guida DVD

Ogni mese Linux Pro vi offre i programmi e le distribuzioni più recenti su DVD

Sul DVD di questo mese...



Giochi e strumenti da installare subito!

» LibreOffice 5.0.2 DEB
» LibreOffice 5.0.2 RPM
» Cube 2: Sauerbraten

Prova i software del confronto p. 50

» ClearOS 7.1
» NethServer 6.6
» TheSSS 13.1
» Zentyal 4.1
» Ubuntu 14.04.3 Server

I migliori programmi selezionati p. 56

» CVAssistant 2.0
» Dust Racing 1.11
» Kid3 3.3
» qBittorrent .2.3
» Qt5CT 0.19
» Scribus 1.5
» Tank of Freedom
» Tmux 2.0
» uGet 2.0.2

Tutto quello che serve per i tutorial p. 66

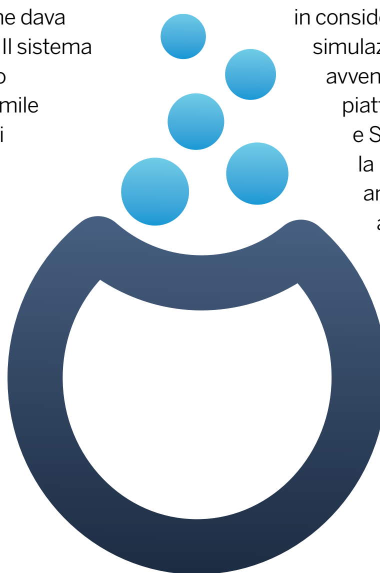
Distro Desktop

Mageia 5

Tra le distro più famose e utilizzate c'è sicuramente **Mageia**. Arrivata alla versione **5**, questa distribuzione basata su Mandriva ha raggiunto lo stato dell'arte. È disponibile sia in versione Live per una prova al volo senza necessità d'installazione sia tramite immagine ISO. Può poi essere scaricata per architetture a 32 bit e 64 bit.

Le principali novità

L'installer è sempre l'affidabile DrakX che permette a chiunque di portare a termine l'installazione in modo semplice e veloce. Durante la procedura, vengono anche scaricati gli aggiornamenti e le dipendenze di cui Mageia ha bisogno per funzionare a dovere. Tra le principali novità della versione 5 troviamo il supporto nativo per le macchine dotate di UEFI. Grub2, inoltre, non è più un componente opzionale, ma fornito di default. Il Centro di Controllo è stato migliorato sia per quanto riguarda le prestazioni sia per la stabilità, infatti sono stati corretti numerosi bug che ne minavano il corretto utilizzo nelle versioni precedenti. Grandi novità anche per quanto riguarda il supporto per i nuovi filesystem. Seppure ext4 rimanga quello predefinito, Btrfs viene adesso supportato come filesystem primario. Come tema standard è stato scelto Adwaita anziché Oxygen-gtk che dava diversi problemi con Gtk+ 3.14. Il sistema di gestione dei pacchetti è stato rafforzato, rendendolo molto simile a quello di Fedora e Suse. Tra gli strumenti disponibili troviamo **ManaTools**, un launcher generico che permette di avviare moduli interni ed esterni. In più, mette a disposizione una vasta serie di strumenti di configurazione che rendono immediata la gestione del sistema anche ai principianti. ManaTools ha poi come obiettivo quello di fornirci un'interfaccia semplice e intuitiva per aggiungere nuovi moduli basati su libYui. Ogni modulo, così come lo stesso ManaTools, può poi essere eseguito tramite interfacce QT, GTK o ncurses. Gli strumenti disponibili sono diversi, tra cui **manaclock** per gestire data e ora, **manadm** per configurare il gestore di login, **manalog** per la lettura dei log di journalct, **manaproxy** per la gestione dei proxy e molti altri. Mageia 5 utilizza il kernel 3.19 capace di supportare una vasta serie di periferiche hardware. Tra queste, le tavolette Wacom dovrebbero essere riconosciute istantaneamente, così come i touchpad Synaptics. I driver proprietari Nvidia340 sono adesso disponibili già in fase di installazione grazie a DrakX. Questo significa che potranno essere rilevate automaticamente le schede grafiche Geforce serie 8xxx e 9xxx. Nel Centro di Controllo è poi stato aggiunto il supporto parziale per NVidia Optimus (ancora non disponibile in fase d'installazione). I desktop supportati sono vari. Troviamo X.Org 1.16.4, KDE 5.1.2. e Gnome. Vengono però anche supportati LXQT, Xfce, LXDE, Cinnamon ed Enlightenment. Le applicazioni per l'uso quotidiano ci sono praticamente tutte, a partire da LibreOffice aggiornato alla versione 4.4. Grande attenzione è stata poi rivolta ai giochi. Troviamo diversi nuovi titoli che prendono in considerazione sport, strategia, simulazione, sparatutto, puzzle, avventura e arcade. In più, troviamo piattaforme di gioco quali Vapor e Steam per Linux. Da sottolineare la presenza di **Lutris**, un nuovo ambiente che mette a disposizione titoli liberi e Open Source per il Pinguino. Se avete già Mageia 4 in esecuzione, è possibile aggiornare senza reinstallare il sistema. Come sempre, consigliamo il backup completo dei dati prima di proseguire: l'aggiornamento da Mageia 3 non è supportato.

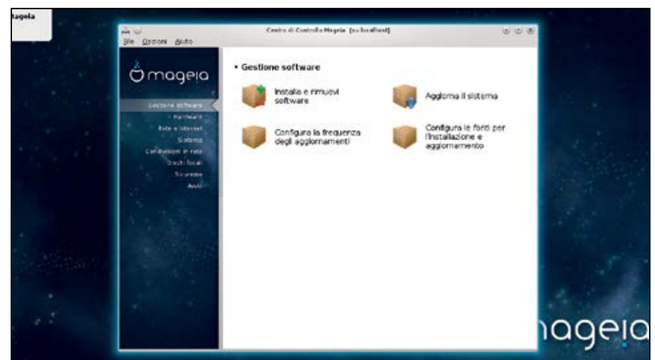


Cosa ci offre Mageia 5



1 Benvenuti in Mageia 5

Al primo avvio la distro ci accoglie con la consueta schermata di benvenuto. Da qui è possibile accedere alle principali funzioni di Mageia, tra cui il Centro di Controllo, la configurazione delle fonti, l'installazione e rimozione del software, nonché l'aggiornamento della distribuzione.



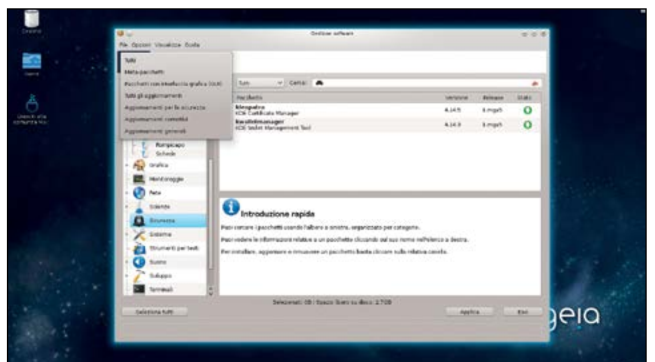
2 Punto focale

Il **Centro di Controllo Mageia** è ancora l'elemento portante della distro. Permette di gestire qualsiasi aspetto in modo semplice e intuitivo. Sulla sinistra è presente il menu delle varie funzioni a cui possiamo accedere: **Software**, **Hardware**, **Rete**, **Sistema**, **Condivisione,** ecc.



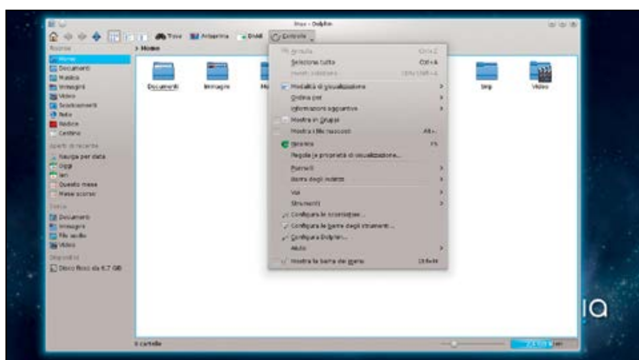
3 Un mondo di applicazioni

Per scaricare i principali software di cui si può aver bisogno fin da subito, è sufficiente premere il pulsante **Applicazioni** nella schermata di benvenuto. I programmi sono suddivisi per categorie accessibili dal menu di sinistra. Se poi non ci accontentiamo, avviamo **RPMDrake** per la gestione dei pacchetti.



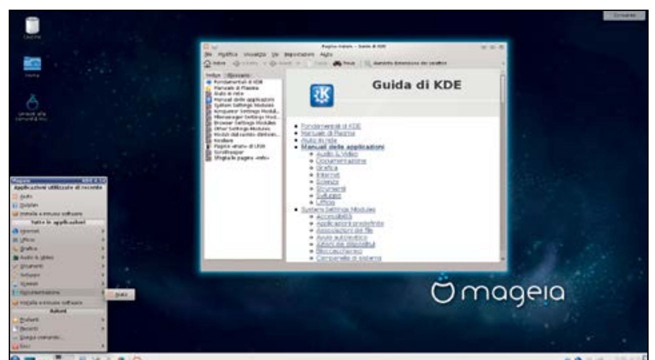
4 Il gestore pacchetti

La gestione dei pacchetti è affidata a **RPMDrake 6.0** che consente di selezionare i vari software in modo semplice e veloce. Possiamo scegliere se includere nella lista quelli dotati di interfaccia grafica (GUI), oppure solo una serie di aggiornamenti per la sicurezza, correttivi o generali.



5 Massimo controllo

Il file manager è **Dolphin**. Permette di sfruttare a pieno le potenzialità di Mageia gestendo tutti gli aspetti relativi ai file, all'archiviazione su disco e alle unità di rete. Particolarmente utile da usare è la funzione **Controllo** che mette a disposizione varie funzioni per ottimizzare l'uso dell'applicazione.



6 Supporto per il desktop

Mageia 5 comprende la documentazione completa di KDE. È accessibile dalla barra delle applicazioni, proseguendo in **Documentazione** e poi **Aiuto**. La guida relativa al desktop varia in base all'ambiente con cui abbiamo deciso di scaricare la distro. LXP

Newsdesk

Ogni mese tutte le novità dal mondo delle aziende e della comunità Open Source

Se hai news da segnalarci o dei commenti scrivi a newsdesk@linuxpro.it

Internet delle cose

Paradiso per hacker

Ormai non sono più al sicuro nemmeno i nostri frigoriferi

Se pensiamo ai dispositivi capaci di mettere a repentaglio la nostra privacy, il frigo che teniamo nell'angolo della cucina non è proprio in cima alla lista degli indiziati. Invece, se possediamo un frigorifero intelligente **Samsung RF28HMELBSR**, il nostro account Gmail potrebbe essere estremamente a rischio. Questo apparecchio ha un display integrato che ci permette di accedere al nostro calendario online, grazie a un account Google. Durante la recente DEF CON hacking conference, si è scoperto che questo frigo non convalida i certificati di sicurezza SSL, quando ci accreditiamo, spalancando così le porte a un possibile attacco e fornendo un accesso potenziale ai nostri dati sensibili. Gli organizzatori dell'evento hanno creato un villaggio IoT in cui c'era anche questo apparecchio, e i partecipanti dovevano hackerarne gli elettrodomestici. Ci sono riusciti i Pen Test Partners che erano inseriti nella stessa rete Wi-Fi. È evidente che si tratta di un caso difficilmente riproducibile nella realtà. Ciò nonostante ha evidenziato che i produttori dovrebbero fare molta più attenzione agli aspetti basilari della sicurezza. Ancora più preoccupante è stata la scoperta di una falla in un portatile **Lenovo**, l'**Y40-80**, da parte dell'utente ge814, nel forum Arstechnica (**http://bit.ly/LenovoG50-80Backdoor**). In pratica, quando si carica Windows 7 o Windows 8, il BIOS del computer controlla **C:\Windows\system32\autochk.exe** per vedere se è il programma creato da Lenovo o quello di Microsoft. Se non si tratta del primo, si sposta su **C:\Windows\system32\0409\zz_sec\autobin.exe** e crea un nuovo **autochk.exe** di Lenovo. Il motivo per cui il produttore vuole che funzioni la propria versione di **autochk.exe** è perché durante il boot, il software di Lenovo scrive **LenovoUpdate.exe** e **LenovoCheck.exe** nella cartella **system32** e li esegue quando è attivo un collegamento a Internet. Non solo Lenovo sta modificando il nostro computer senza autorizzazione ed esegue programmi "telefono casa" sfruttando il nostro collegamento Internet, ma ge814 ha anche scoperto che questi programmi non usano SSL, il che mette di nuovo in pericolo i nostri dati. Siccome questo problema colpisce Windows, se proviamo a sostituire quel sistema operativo con Linux, allora il BIOS ci creerà un sacco di grattacapi. Di fronte a tutto ciò, Linus Torvalds ha voluto però ricordare che: "L'unica vera soluzione per la sicurezza è ammettere che gli errori accadono" e che "Per ridurli, è necessario un controllo a più livelli, dove quello successivo controlla gli errori del precedente". LXP

> "L'unica vera soluzione per la sicurezza è ammettere che gli errori accadono"



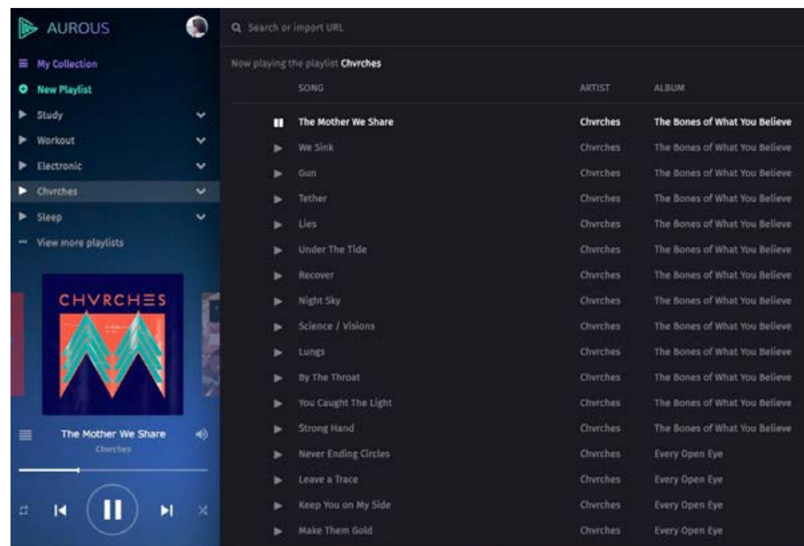
› **Non facciamoci ingannare dalla sua aria innocente, perché questo frigorifero, tecnologicamente avanzato, potrebbe diventare un cavallo di Troia per hacker!**

Torrent

Musica gratis in streaming

Così si riproducono i file musicali tramite un client Torrent, ma è legale?

Prendiamo un pezzetto del codice genetico di **Popocorn Time**, il client **Torrent** per lo streaming video, e uniamolo a quello di **Spotify**, il noto servizio di streaming audio e otterremo una nuova creatura che ci permetterà di riprodurre i file musicali dalla rete Torrent, senza doverli nemmeno scaricare. La creatura di cui stiamo parlando si chiama **Aurous** (**https://aurous.me**), una nuova applicazione multipiattaforma, che funziona indifferentemente su Windows, Mac OS X e i sistemi operativi Linux. La sua interfaccia utente deve molto al noto servizio di streaming musicale, ma la sua vera forza sta nel cuore, cioè nella sua capacità di gestire in streaming i file Torrent. Per trovare ciò che ci interessa, possiamo scrivere il nome dell'artista, oppure del brano, o dell'album in cui si trova. Oltretutto non dobbiamo nemmeno preoccuparci dei formati audio, poiché Aurous è capace di leggerne moltissimi, tra cui WAV, FLAC, OGG, OPUS e perfino WebA. Un'altra buona notizia per gli utenti più smaliziati è la possibilità che gli sviluppatori ci concedono di personalizzare il client tramite API e plug-in, in modo da ottenere il massimo. La storia di Aurous è abbastanza anomala, perché comincia con un'operazione di crowdfunding che però non è andata a buon fine. Tuttavia, i suoi creatori non si sono affatto scoraggiati e sono riusciti a portare a termine l'impresa con risultati apparentemente ottimi. Attualmente però la magistratura ha bloccato tutti gli scaricamenti del client e nel giro di una settimana deciderà se consentirne la ripresa. Sarà quindi legale fare uso di questa app? A breve la risposta... LXP



› Aurous è una potente applicazione multipiattaforma, funzionante con Windows, Mac OSX e Linux, in grado di eseguire direttamente un file audio tramite Torrent, senza doverlo scaricare nel nostro disco fisso

Android

Android 6.0 sui Nexus

I marshmallow sbarcano sui nuovi dispositivi Android e su altri ancora...

Nel momento in cui scriviamo Google ha iniziato la distribuzione via OTA della nuova versione di **Android**, la **6.0 Marshmallow**. I dispositivi interessati sono i Nexus 5, 6, 7 (2013) e 9, oltre naturalmente ai due nuovi modelli appena presentati. Non viene così più aggiornato il Nexus 4 ma a tre anni di distanza dal debutto questo è abbastanza normale.

Tra le principali novità del nuovo sistema oltre alla gestione nativa dell'impronta digitale spicca il sistema di gestione della batteria **Doze** che ridurrà al minimo i consumi quando il dispositivo non viene utilizzato per molto tempo. Inoltre i permessi per le singole app non verranno concessi tutti insieme al momento dell'installazione ma solo quando servono effettivamente. LXP



› Google continua la tradizione di fornire alle nuove versioni del sistema il nome di un dolce tipico, per Android 6.0 è *Marshmallow*, un dolcetto morbido molto zuccherato

LE MIGLIORI DISTRO 2015

Il 2015 è quasi finito ed è quindi il momento di fare un bilancio: qual è la migliore distro dell'anno?

L'universo Linux è pieno di distribuzioni e ognuna di queste meriterebbe sicuramente un po' di attenzione. Ciò non toglie che ce ne siano alcune migliori di altre. Una distro è molto più della somma delle sue parti. Alle sue spalle c'è sempre il lavoro di un team di appassionati che dedicano anima e corpo a realizzare qualcosa che possa lasciare il segno. C'è però un aspetto più importante da valutare: la differenza tra le distribuzioni generaliste e quelle specializzate. Le prime sono fatte per soddisfare le esigenze dei molti, basandosi soprattutto sull'uso di un desktop specifico e un approccio quanto più diretto possibile. Basta pensare a **OpenSUSE** come diretta interpretazione del desktop KDE, oppure **Fedora** con GNOME e ancora **Ubuntu**.

Le distro specializzate, invece, sono fatte per dare il loro meglio in un ambito particolare e mirato. Spesso, quindi, non sono adatte all'uso quotidiano, quanto più a un'attività particolare: sicurezza, multimedia e via dicendo. Naturalmente, un altro elemento che distingue questi due macro-insiemi è la velocità con cui si evolvono. Le piattaforme come Ubuntu che hanno alle spalle solide multinazionali e programmatori stipendiati possono contare su aggiornamenti frequenti. Altre, come per esempio **Linux Mint**, pur basandosi su una schiera di appassionati, riescono a tenere il passo delle controparti più fortunate dal punto di vista del sostegno economico. Altre ancora, invece, devono fare i conti con la disponibilità del proprio team e quindi procedono più lentamente. In tutti i casi, il software Open Source viene sempre in aiuto, consentendo a ogni progetto di svilupparsi indipendentemente dalle proprie risorse. Nelle prossime pagine affronteremo quindi una serie di valutazioni che avranno come obiettivo l'analisi di varie distribuzioni. Attenzione: abbiamo volutamente ignorato le versione appena uscite, come per esempio Ubuntu 15.10, perché non c'è stato ancora tempo di provarle estensivamente e abbiamo quindi dato la precedenza a versioni con più mesi di vita, e di test, alle spalle.

> "Le distro più popolari mirano a offrire un'esperienza desktop stabile, pratica e funzionale"

I contendenti

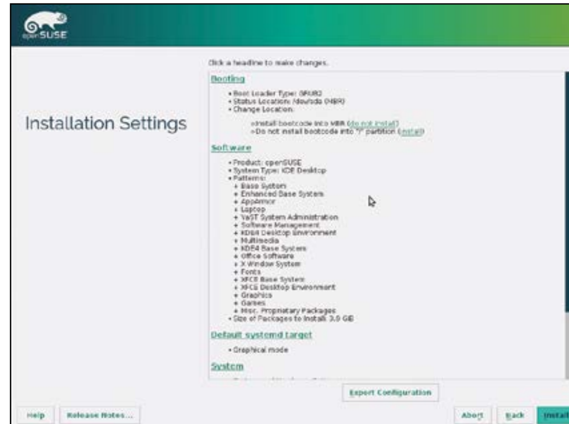
Mageia 5 ■ Ubuntu 15.04 ■ Linux Mint 17.2 ■ Fedora Workstation 22 ■ OpenSUSE 13.2

Installazione e aggiornamento

Partiamo subito dalle basi...

Seppure alcuni produttori come Dell e Lenovo abbiano aderito all'offerta di PC con Linux preinstallato, molti principianti devono ancora confrontarsi con l'installazione di una distro. Indipendentemente dal computer che si utilizza, l'uso di una distribuzione semplice da installare può fare la differenza rispetto ad altre soluzioni.

Fedora, per esempio, ha revisionato l'installer **Anaconda** che adesso si avvale di una procedura molto più flessibile. Seppure non si possa ancora definire il massimo in fatto di semplicità, ha raggiunto un buon livello di sviluppo che gli conferisce un'ottima stabilità. Gli utenti avanzati possono utilizzare Anaconda per creare partizioni LVM, ma a differenza di altri installer non è ancora possibile eseguire un aggiornamento a una nuova versione. Ciò non toglie che il nuovo strumento di **FedUp** della distro sia in grado di utilizzare un repository di rete o un'immagine DVD come sorgente di un pacchetto. Uno degli installer più semplici da gestire è quello di Ubuntu che peraltro viene preso in prestito anche da altre distro, tra cui Linux Mint. La procedura è semplice e intuitiva anche per i nuovi utenti. La versione originale dell'installer possiede poi una serie di opzioni utili ad aggiungere aggiornamenti e software di terze parti. Queste opzioni, tuttavia, non sono presenti nella release disponibile in Mint. Essa, infatti, provvede all'installazione automatica di codec e plug-in senza il nostro intervento. Infine, avete la possibilità di partizionare il disco sfruttando LVM e la cifratura. In ogni caso, anche tale installer non è stato progettato per aggiornare la distro. In Ubuntu, questo specifico aspetto, viene gestito da **Update Manager** il quale controlla la disponibilità di nuove versioni e vi aiuta a eseguire il processo di aggiornamento. Mint, invece, preferisce un approccio più radicale: consiglia ai propri utenti di procedere con una nuova installazione. Ciò non toglie che, sfruttando lo strumento **MintUpdate**, sia ugualmente possibile eseguire un aggiornamento senza reinstallare il sistema. L'attuale ramo 17.x di Mint è una versione LTS che consente di ricevere aggiornamenti riguardanti la sicurezza e correzioni di vario genere fino ad aprile 2019. OpenSUSE e Mageia possono invece contare su due installer tra i più navigati. Entrambe dispongono di DVD per l'installazione che hanno un peso di circa 4 GB, al cui interno è però presente una nutrita schiera di software. Sono disponibili vari desktop, tra cui KDE, Gnome, Xfce e LXDE. **Mageia**, in più, aggiunge **Cinnamon** e **Mate**. L'installer di OpenSUSE consente poi il partizionamento LVM e la cifratura del disco e permette di creare gli utenti già durante la fase d'installazione. Inoltre, è l'unico che consente di selezionare un sistema di autenticazione per la rete, come LDAP o NIS, nonché un protocollo di crittografia delle password. Il partizionatore presente nell'installer di Mageia può essere usato in modalità semplice o avanzata. La prima crea un layout di base, mentre la seconda permette di separare le partizioni secondo l'uso che vogliamo farne. Mageia, infine, è l 'unica distro che permette di scegliere il bootloader, supportando **Grub**, **Grub 2** e **Lilo**. Infine, è possibile installare la piattaforma su macchine UEFI e rivedere in un secondo momento le impostazioni scelte durante l'installazione.



› L'installer di *OpenSUSE* consente di salvare la configurazione corrente all'interno di un file XML. Questo può poi essere riutilizzato per altre installazioni

Verdetto

Fedora Workstation 22
★★★☆☆

Linux Mint 17.2
★★★★☆

Mageia 5
★★★★★

OpenSUSE 13.2
★★★★★

Ubuntu 15.04
★★★★☆

» Gli installer di Mint e Ubuntu non sono altrettanto flessibili come quelli di Mageia e OpenSuse.

»

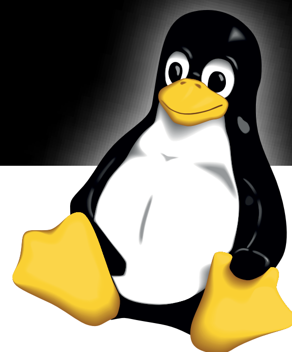
Distro specialistiche

Oltre alle distro più conosciute, del mondo Linux fanno parte anche sistemi specializzati per determinati compiti. Un esempio è **OpenMediaVault**. Basata su Debian, è perfetta per convertire un vecchio computer in un server NAS. Allo stesso modo, **IPFire** è pensato per riutilizzare un PC oramai obsoleto come firewall hardware. Da segnalare il progetto **TurnKey** che realizza applicazioni JeOS per le distro server specializzate nella gestione dei contenuti e nello sviluppo Web. Da non dimenticare **Kali Linux** che contiene centinaia di strumenti per l'analisi forense, inclusi svariati programmi per mettere alla prova la sicurezza di qualsiasi infrastruttura informatica.

Se invece amate la privacy, vi consigliamo di dare un'occhiata a **Linux Tails**. Questa distro mette a disposizione un gran numero di applicazioni preconfigurate per mantenere l'anonimato in Rete. Tails sfrutta la rete Tor e include software per la crittografia utile a cifrare file, email e messaggi istantanei. C'è poi **SteamOS**, la distro basata su Debian e prodotta da Valve, pensata per i giocatori incalliti. Se però volete andare oltre e cimentarvi anche con titoli non presenti sulla piattaforma di Valve, allora puntate su **Play Linux**. Si tratta di una distro basata su Ubuntu che getta le basi per offrire un ambiente ideale per il divertimento.



› *Tails*, oltre ad assumere le sembianze di Windows, comprende anche il client bitcoin *Electrum*



Esperienza d'uso

Come si comportano all'atto pratico?

Visto che la maggior parte delle distro comprende le stesse applicazioni, ciò che più fa la differenza è l'esperienza d'uso. Gli sviluppatori dedicano parecchio tempo ad apportare una serie di modifiche che permettano agli utenti di utilizzare la propria piattaforma in modo soddisfacente. Tutte le distribuzioni, infatti, si sforzano di apparire come unità complete e coese e non come semplici accozzaglie di software diversi. Tra le attività che in questo senso richiedono più risorse, c'è proprio l'integrazione delle varie app con il desktop. Vediamo quindi come si comportano in questo contesto i nostri concorrenti.

Mageia 5 ★★★★★



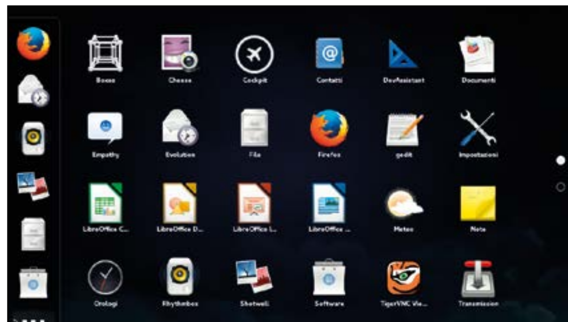
Mageia è una distro che si distingue fin da subito per la sua eleganza. Il tema personalizzato su desktop KDE mette subito a proprio agio anche i più inesperti. Mageia, poi, ha sempre puntato molto sull'esperienza d'uso rivolta a tutti e anche questa versione non fa eccezione. La distro vi accoglie con un'applicazione di benvenuto che, a differenza di quelle viste in altre piattaforme, mette a disposizione diverse funzioni. In primo luogo vi informa sullo stato dei repo e in più consente di installare alcuni programmi Open Source di uso comune. Oltre al supporto su DVD, Mageia può essere utilizzata in modalità Live con desktop Gnome. La distro ha poi una serie di strumenti estremamente personalizzabili. In più, mette a disposizione diverse utility che possono essere utilizzate sia dai principianti sia da chi ha un po' più di esperienza sulle spalle.



Fedora Workstation 22 ★★★☆☆



I problemi di usabilità con Fedora iniziano fin dall'installazione. Seppure il processo sia ben progettato, non è altrettanto definito. Il desktop Gnome 3 non riesce a essere particolarmente incisivo e lascia una strana sensazione di vuoto. Gli utenti che non sono abituati a questo ambiente dovranno prima di tutto confrontarsi con alcuni limiti. Tra questi, l'impossibilità di posizionare icone o cartelle su desktop e l'assenza di una barra degli strumenti disposta nella parte inferiore. La visualizzazione delle applicazioni non è così efficace come la vista per categoria che viene usata da altre distro simili. Detto questo, Gnome 3.16 presenta numerosi miglioramenti, tra cui un nuovo sistema di notifica. L'attenzione di Fedora si concentra soprattutto sull'interazione di diversi ambienti desktop, cercando così di soddisfare le esigenze di più persone.



App preinstallate Bastano per soddisfare tutte le esigenze?

Ubuntu, Mint e Fedora mettono a disposizione CD live, mentre Mageia e OpenSUSE aggiungono anche supporti d'installazione solo DVD. Tutte queste distro gestiscono poi più ambienti desktop. Tuttavia, nessuna consente la selezione dei pacchetti. OpenSUSE e Mageia sono comunque le migliori per flessibilità. Una volta scelto l'ambiente, potete selezionare i gruppi di software per le varie funzioni (ufficio, multimedia, gioco e via dicendo). In più, entrambe le distro permettono di installare pacchetti specifici per server Web, database, o gateway firewall. Infine, è possibile utilizzare gli installer per ottimizzare la selezione dei pacchetti e scegliere i singoli da aggiungere. Al di là di questo, tutte le distro prese in considerazione comprendono le solite applicazioni per l'uso quotidiano, tra cui i tradizionali LibreOffice e Firefox. Alcune piattaforme richiedono comunque l'aggiunta di codec o specifici plug-in ai browser per riprodurre contenuti flash. Ubuntu vi permette di aggiungere questo materiale durante il processo d'installazione. Mint, invece, lo fa in automatico. Mageia, Fedora e OpenSUSE, infine, dispongono di un processo ben documentato per l'aggiunta dei componenti aggiuntivi.

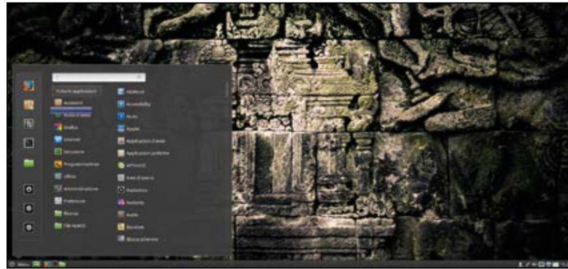
Verdetto

Fedora Workstation 22 ★★★☆☆
Linux Mint 17.2 ★★★★☆
Mageia 5 ★★★★☆
OpenSUSE 13.2 ★★★★☆
Ubuntu 15.04 ★★★★☆

» *Tutte le distro hanno una selezione di applicazioni simili.*

Linux Mint 17.2 ★★★★★

Mint è una delle distribuzioni che ha riscosso più successo. È riuscita a scalare le classifiche di **DistroWatch** (**http://distrowatch.com**), grazie al sapiente mix composto da semplicità d'uso, flessibilità e usabilità. Basata su Ubuntu, Mint modifica l'approccio del concorrente mettendo a disposizione una serie di strumenti molto più accessibili e diretti. Tra questi, non si può fare a meno di menzionare il desktop **Mint Software Manager** che può essere paragonato all'omonimo **Ubuntu Software Center**. Il punto di forza di Mint rimane comunque il desktop **Cinnamon**. Basato su **Gnome 3**, conserva l'aspetto della versione 2. L'ambiente è immediato e mette a proprio agio anche chi non ha mai utilizzato una distro Linux prima.





Ubuntu 15.04 ★★★★☆

Ubuntu si distingue senza dubbio per il suo installer. Infatti, è uno dei più semplici da utilizzare. Consente di gestire a dovere lo spazio a disposizione, nonché configurare una gestione dual-boot del PC in poche e semplici mosse. Ciò che invece lascia perplessi è il desktop. Unity, anche se non sembra poi così tanto diverso rispetto a Gnome 3, può disorientare chi è abituato ai tradizionali ambienti di lavoro. Per utilizzarlo al meglio è infatti necessario acquistare un po' di familiarità con gli strumenti principali. Una volta preso il controllo del desktop, però, si può iniziare ad apprezzarne l'ottima integrazione con le applicazioni. Il menu che riepiloga lo stato dei diversi account di messaggistica istantanea è uno dei migliori in circolazione. Se poi ci aggiungiamo altre applicazioni davvero ben realizzate come **Ubuntu Software Center**, ecco che si raggiunge un'eccellente esperienza d'uso.

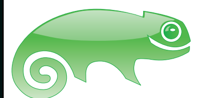




OpenSUSE 13.2 ★★★☆☆

Uno degli aspetti più piacevoli di questa distro è la sua capacità di personalizzare le applicazioni di base per meglio integrarsi con il desktop. OpenSUSE, inoltre, tende a modificare alcuni aspetti caratteristici di KDE per garantire la massima usabilità con il sistema. La distro sfrutta tutti gli strumenti più comuni già visti in altri ambienti, fatta eccezione per Yast che si occupa di installare e configurare OpenSUSE. Sebbene non ci sia da ridire niente sulla maggior parte delle funzioni, alcune potrebbero essere riviste e aggiornate. Il gestore pacchetti, per esempio, non può essere minimamente paragonato a **Ubuntu Software Center**. OpenSUSE fa un ottimo lavoro su tutti i fronti. In più, potete contare su un'eccellente documentazione e un supporto di prim'ordine. Tuttavia non è una piattaforma adatta ai principianti come Ubuntu o Mint.





Gestione pacchetti Aggiungere e rimuovere gli strumenti

Per quanto una distro sia equipaggiata con una vasta schiera di software, prima o poi vi troverete a fare i conti con il gestore pacchetti.
Ogni piattaforma ne possiede uno che può essere a riga di comando o grafico. La versione 22 di Fedora ha segnato l'arrivo di **DNF** che va così a sostituire Yum. Ubuntu, invece, si pone al primo posto grazie al suo **Software Center**, uno dei migliori in circolazione, fornito di una gran quantità di applicazioni, semplice da usare e configurare. Come la maggior parte dei gestori pacchetti, elenca solo quelli contenuti nei repo ufficiali. Tuttavia basta accedere alla funzione **Software e Aggiornamenti** per aggiungere o rimuovere i repo da cui prelevare i software.
Mint, seppure si basi su Ubuntu, ha deciso di utilizzare un gestore pacchetti diverso. Il suo **Software Manager**, pur avendo funzioni simili a quelle della distro madre, fornisce una serie di opzioni di vario genere. La piattaforma include poi **MintSources**, uno strumento per la gestione dei sorgenti software, nonché il gestore pacchetti **Synaptic** utile per gli utenti più esperti. Mageia e OpenSUSE sfruttano i rispettivi centri di controllo basati su pacchetti RPM. OpenSUSE fa riferimento a **Zypper**, mentre Mageia punta su URPMI. Quest'ultimo, seppure non abbia una grafica piacevole, è molto funzionare e abbastanza intuitivo.

Verdetto

Fedora Workstation 22
★★★☆☆

Linux Mint 17.2
★★★★☆

Mageia 5
★★★★★

OpenSUSE 13.2
★★★★☆

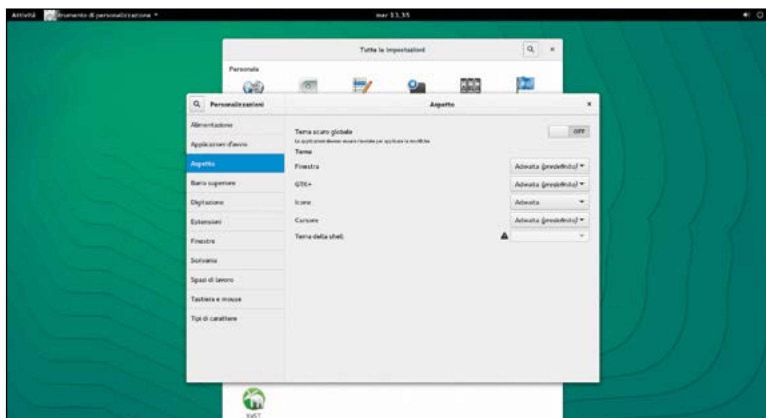
Ubuntu 15.04
★★★★☆

» *Mageia ha un leggero vantaggio grazie alla sua flessibilità.*

Opzioni di configurazione

Quanto possiamo spingerci nella personalizzazione?

Gli utenti Linux sono sempre stati in grado di plasmare una distro in base al tipo di utilizzo che ne fanno. Tuttavia il grado di controllo varia da una piattaforma all'altra. Alcune, come Ubuntu, non forniscono molti strumenti per la personalizzazione. Per l'occasione, avrete quindi bisogno di software di terze parti come Unity Tweak Tool. Fedora non è molto diverso. La distro non dispone di un proprio pannello per le impostazioni, bensì si basa su quello fornito da Gnome. Quest'ultimo è molto diverso da quello di Ubuntu sia in termini di opzioni, sia per quanto riguarda la configurazione in generale. Mint, a differenza di Fedora, raggruppa tutte le funzioni per la personalizzazione in un unico strumento che consente di modificare l'aspetto del desktop e ottimizzare il compositing. Una variante fondamentale rispetto a Ubuntu sta nell'uso della funzione **Device Driver**. Questa utilizza un'interfaccia utente ottimizzata, capace di consentire agli utenti la scelta di un determinato driver in base al dispositivo. OpenSUSE e Mageia, invece, dispongono di ampi pannelli di controllo che permettono di modificare le impostazioni scelte in fase di installazione. Yast di OpenSUSE, per esempio, si rivolge sia agli utenti desktop sia agli amministratori di sistema. Questo strumento consente di modificare tutte le impostazioni di una normale configurazione desktop: bootloader, firewall, gestione degli utenti, configurazione di rete, regole di sicurezza, servizi di sistema, ecc. In più, funziona come gestore pacchetti. Il centro di controllo di Mageia offre strumenti molto simili. Gli utenti esperti possono creare una VPN e molto altro. Il team di Mageia, inoltre, sta lavorando a un nuovo centro di controllo chiamato **ManaTools**, incluso in anteprima in Mageia 5.



› Grazie ai loro Centri di Controllo, Mageia e OpenSUSE hanno i migliori strumenti di configurazione

Verdetto

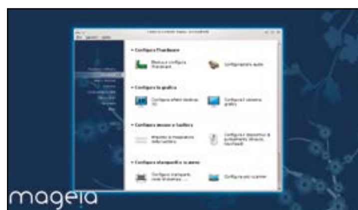
Fedora Workstation 22
★★★☆☆
Linux Mint 17.2
★★★★☆
Mageia 5
★★★★★
OpenSUSE 13.2
★★★★★
Ubuntu 15.04
★★★☆☆

» Le funzioni che distinguono Mageia e OpenSUSE dalle altre distro sono nei rispettivi pannelli di controllo.

Desktop predefiniti

Districhiamoci tra le funzioni

Molte distro permettono di scegliere vari desktop oltre a quello predefinito. Ubuntu, sotto questo aspetto, è un'eccezione. Infatti supporta solo Unity e non intende fare spazio a nessun'altra alternativa. Questo ambiente, seppure sia migliorato con il tempo, è ancora troppo diverso rispetto ai principali concorrenti. Se però siete disposti ad affrontare le sue peculiarità, potreste anche trovarlo piacevole. Fedora è considerato il fiore all'occhiello di Gnome. La versione 3 di questo desktop è però ancora più spiazzante rispetto a Unity. Infatti, a differenza di altre distro basate sullo stesso ambiente, Fedora ha deciso di non modificare alcun aspetto di Gnome. Dovrete quindi trascorrere del tempo a ottimizzarlo secondo le vostre esigenze. OpenSUSE e Mageia, pur potendo utilizzare Gnome come ambiente secondario, preferiscono puntare su KDE. Quest'ultimo si basa sulla classica scrivania ed è quello più adatto a un approccio semplificato. Ciò non toglie che per sfruttarlo al meglio sia necessario prendere una certa confidenza con gli strumenti principali messi a disposizione. Trascorso un periodo di rodaggio, non farete fatica ad esaltarne i tanti aspetti positivi per cui si distingue. L'unico fattore negativo di KDE è la carenza di opzioni modificabili. Grazie a Cinnamon, Mint può essere considerato tra le migliori distro in ambito desktop. L'ambiente è ben strutturato, intuitivo e pratico da utilizzare sia per gli utenti alle prime armi sia per chi ha più esperienza sulle spalle.



› Grazie al Centro di Controllo di Mageia, si può configurare il server grafico per la gestione degli effetti 3D

Verdetto

Fedora Workstation 22
★★☆☆☆
Linux Mint 17.2
★★★★☆
Mageia 5
★★★★☆
OpenSUSE 13.2
★★★★☆
Ubuntu 15.04
★★☆☆☆

» Ubuntu e Fedora rimangono indietro a causa dei desktop non destinati a un uso immediato.

Distro per principianti

Tutto quello che serve per iniziare

Elementary OS "Freya"

Questa piattaforma è basata su **Ubuntu**, anche se all'atto pratico ha davvero poco in comune con la distro madre. Sfrutta una proprio desktop chiamato **Pantheon** con diverse applicazioni personalizzate, tra cui un dock simile a quello di Mac OS X. La distribuzione pone particolare attenzione alla grafica, ribadendo il concetto con continui richiami al mondo Mac. Tra gli strumenti che possiamo trovare, abbiamo **Mail Geary**, l'editor di testo **Scratch** e il player video **Audience**. Tutte queste applicazioni sono progettate per l'uso da parte di utenti alle prime armi. Elementary OS usa il proprio windows manager chiamato **Gala** che ha il pregio di consumare davvero poche risorse.



Korora 22

Korora è basata su **Fedora**, ma separa le versioni Gnome da quelle Live che sfruttano KDE. Rispetto alla distro madre, Korora si distingue per la possibilità di personalizzare il desktop in ogni suo aspetto. Tra i vari strumenti presenti troviamo anche **Gnome Tweak Tool** che permette di mettere mano alle impostazioni di Gnome in modo semplice e veloce. La distro dispone poi di pieno supporto per i file multimediali e può contare su svariati software di terze parti, come **RPMFusion**, **Google Chrome** e **VirtualBox**. Korora utilizza anche altre applicazioni popolari come **Audacity**, **OpenShot** e **Handbrake**. In più aggiunge al browser Firefox una serie di estensioni predefinite. Per quanto riguarda la gestione dei pacchetti, Korora si affida a **YumExtender**.



Pinguy OS 14.04.2

Un'altra distro che ha il pregio di attirare nuovi utenti è PinguyOS. Gnome, pur essendo il desktop predefinito, può essere personalizzato con un dock particolarmente vivace. In Pinguy troviamo poi **Ubuntu Tweak Tools** per la modifica di svariati parametri operativi. La distro è piena di programmi di vario genere, tra cui il server multimediale **Plex**. Oltre ai tanti software Open Source presenti, troviamo anche qualche applicazione proprietaria come **TeamViewer**, **Spotify** e **Steam** per Linux. C'è anche **Wine** che potete gestire tramite il front-end predefinito **PlayOnLinux**. Se avete bisogno di più software, potete comunque rivolgervi a Ubuntu Software Center e al gestore pacchetti Synaptic. La distro, infine, utilizza i propri repo oltre a quelli per Ubuntu e Mint.

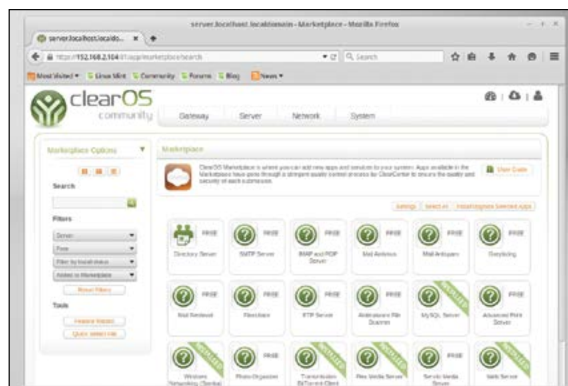


Il verdetto Distro per principianti

Tutte e tre le distribuzioni desktop che abbiamo valutato si sforzano di proporsi al meglio. Di base forniscono molti programmi, ambienti desktop pratici, intuitivi e allo stesso tempo configurabili e mettono a disposizione parecchio software supplementare.
Delle tre, Elementary OS è però quella che più si distingue. Tutto, dal gestore applicazioni fino ai vari menu, è realizzato in modo da aderire perfettamente ai suoi principi di funzionamento. C'è poi Korora che ha trasformato le basi di Fedora in un ambiente completamente funzionale. Questa distro, infatti, si dimostra un ottimo punto di partenza per chiunque voglia addentrarsi nel mondo Linux. In più, ha un'ottima capacità di personalizzazione. Il punto debole di Korora è però l'installer **Anaconda** ereditato da Fedora. PinguyOS presenta fin da subito un ottimo mix tra forma e funzioni. L'ambiente desktop permette di accedere fin da subito a un vasto numero di applicazioni. Assicuratevi solo di avere una macchina abbastanza potente. Le varie modifiche che si possono apportare al sistema hanno un proprio peso e in linea di massima vanno a inficiare sulle prestazioni. Il quantitativo minimo di RAM per utilizzare con soddisfazione PinguyOS è di almeno 4 GB. In caso contrario, è meglio passare a **ElementaryOS**.

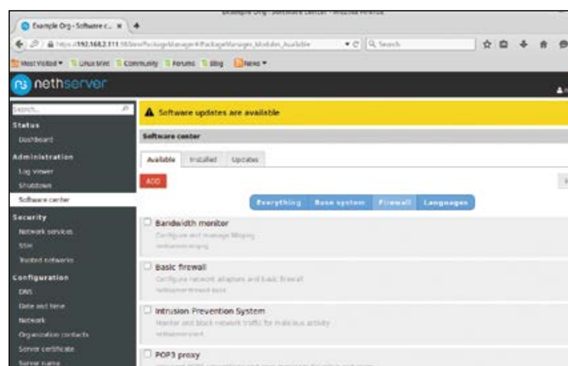
Distro per server

Per chi pretende di più



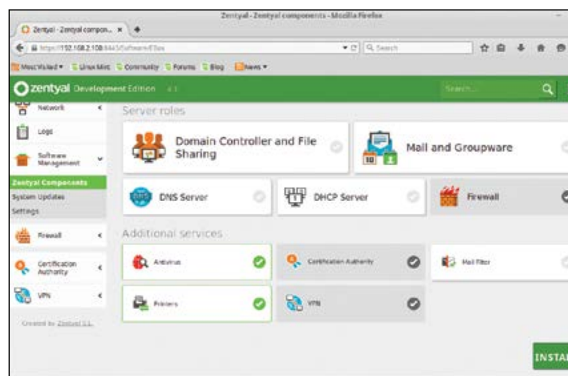
ClearOS 6.6

Uno dei maggiori vantaggi di una distro basata su **CentOS** consiste nella vasta disponibilità di repository supportati. ClearOS permette di creare un server personalizzato in base al suo utilizzo: rete protetta (ufficio), rete aperta e accessibile o gateway. La distribuzione supporta oltre 80 servizi gratuiti per varie funzioni, tra cui un server di rete, per la Cloud e molto altro ancora. È poi possibile usarla come seedbox o Plex Media Server. ClearOS include anche diversi strumenti di sistema che si uniscono a quelli per la gestione della rete.



NethServer 6.6

Anche alla base di NethServer troviamo **CentOS**. Questa distro consente di gestire il proprio server tramite una comoda interfaccia Web accessibile via browser. NethServer dispone poi di una serie di software personalizzati che possono essere aggiunti in base alle necessità. Sfruttando queste applicazioni, configuriamo un firewall, un file server, un server Web e molto altro ancora. La dashboard di NethServer basata su browser è ben strutturata. Ogni sezione dispone di un pulsante di supporto che spiega le varie opzioni. Da qui è possibile ottenere una panoramica sui vari parametri del server installato. La dashboard, inoltre, include un visualizzatore di log per il monitoraggio dei registri di tutti i servizi presenti. Oltre alla documentazione in linea, sul sito Web troviamo anche una guida molto dettagliata.



Zentyal 4.1

A differenza delle altre due distribuzioni basate su **RPM**, Zentyal prende a modello Ubuntu Server. Questa distro utilizza un desktop minimale, ma consente di accedere a un'interfaccia di gestione Web. Si possono poi prendere i comandi direttamente da remoto per configurare l'installazione. Rispetto alle altre due soluzioni sopra citate, Zentyal non è un server adatto a tutti gli usi, bensì si concentra sull'utilizzo negli uffici. Infatti, è possibile sfruttare questa distro per creare un server capace di filtrare le email, scansionare i sistemi alla ricerca di virus, gestire le stampanti, le VPN e gli altri servizi tipici di un ufficio. Una volta installato, Zentyal consente di configurare le varie funzioni dall'interfaccia Web. Quest'ultima si distingue per la struttura ordinata e precisa, capace di mettere subito a disposizione gli strumenti più utili. La distro, infine, pur non disponendo di una funzione per installare un server Web, può usare Apache.

Il verdetto Distro per server

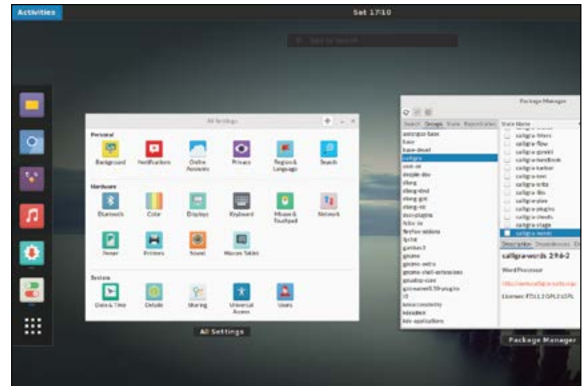
Distribuire e configurare un server è un processo particolarmente difficile. Le tre distro che abbiamo proposto si distinguono soprattutto per comodità e flessibilità. In più permettono di creare piattaforme complesse utilizzando un'interfaccia Web "punta e clicca". Tutti e tre i sistemi hanno comunque una curva di apprendimento poco ripida e consentono di scegliere una vasta serie di server da utilizzare. In definitiva, sono relativamente simili tra loro. Ignorando le differenze minori, possiamo dire che ognuno offre la stessa esperienza d'uso sia nella gestione sia nella configurazione dei rispettivi componenti. Ciò su cui basiamo il nostro giudizio consiste nel valutare il numero di server e i servizi offerti. Zentyal, da questo punto di vista, non brilla particolarmente. Va comunque detto che si tratta di una soluzione ideale per chi vuole un server da usare in ufficio. ClearOS, invece, è agli antipodi: offre una grande quantità di servizi e permette di mettere in piedi uno specifico server in pochi passaggi. Purtroppo, però, non è adatto a tutti i tipi di distribuzioni. Se per esempio si vuole lavorare con **OwnCloud**, è meglio puntare su **NethServer**. Se siete amministratori vecchia scuola, probabilmente vorrete costruirvi il vostro server da zero. In questo caso, consigliamo di puntare su **Ubuntu Server** o **CentOS** in base alla preferenza per i rispettivi gestori pacchetti.

Rolling release

Per chi ama il rilascio continuo

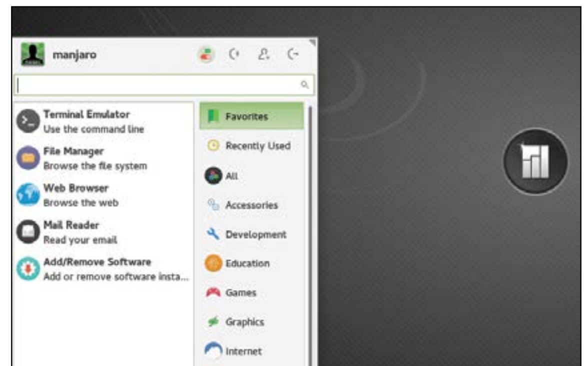
Antergos

Antergos è una distro rolling release basata su **Arch Linux**. Utilizza i repo ufficiali di Arch, ma può contare anche sui propri. Ufficialmente, questa piattaforma utilizza una versione modificata di **Gnome**. Ciò non toglie che sia possibile sostituirla con KDE, Cinnamon, Mate, Openbox o LXDE. Per impostazione predefinita, Antergos viene fornito con il browser **Chromium** dotato di plug-in Flash. Tuttavia, durante l'installazione è possibile aggiungere **Firefox**, così come qualche altro software che non trovate per impostazione predefinita (**LibreOffice** è un esempio). Il gestore pacchetti è **PacMan** di **Arch**.



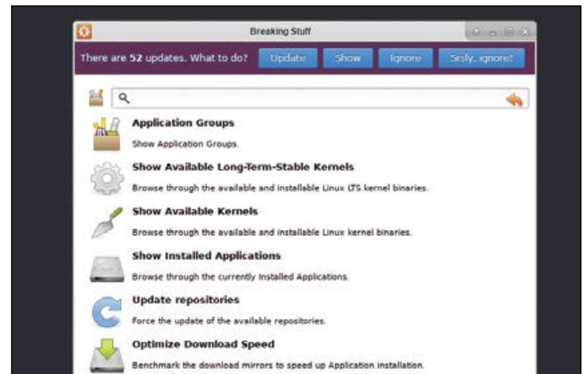
Manjaro 0.8.13.1

Manjaro è un'altra distro rolling release basata su **Arch Linux**. L'installer è simile a quello utilizzato da Antergos. A differenza di quest'ultima che sfrutta Gnome come desktop predefinito, Manjaro preferisce **Xfce**. Esistono comunque versioni con Gnome, KDE, Cinnamon, Mate ed Enlightenment. In Manjaro trovate una funzione per la gestione personalizzata delle impostazioni che tuttavia si limita a offrire lo stretto indispensabile. L'unico aspetto degno di nota è la possibilità di utilizzarla per installare facilmente un kernel diverso da quello predefinito. Le applicazioni non mancano. Troviamo **Firefox**, **LibreOffice**, **VLC** e **Steam** per Linux. Come Antergos, Manjaro si affida ai repo di **Arch**, sfruttando poi **AUR** e **PacMan**.



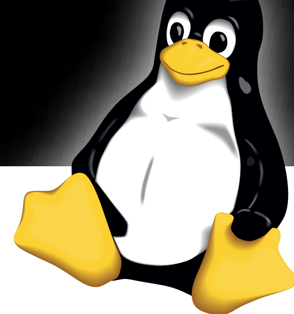
Sabayon 15.07

Sabayon è un'altra distro rolling release estremamente ben fatta. Basata sul ramo testing di **Gentoo**, conserva l'anima del suo genitore, ma si distingue soprattutto per la semplicità d'uso. Si dimostra quindi una piattaforma adatta anche ai principianti. Ci sono diverse varianti basate su vari desktop, come Gnome, KDE e Xfce. Per l'installazione, Sabayon utilizza una versione altamente modificata di **Anaconda** che si trasforma così in uno strumento semplice da usare. Tra le applicazioni incluse, ne troviamo alcune proprietarie come **Google Crome**. Ci sono poi **Gimp** e **LibreOffice**. La gestione dei pacchetti è affidata all'ottimo **Rigo Application Browser** che colpisce subito per flessibilità e praticità. È possibile utilizzare Rigo per installare e aggiornare le singole applicazioni, nonché per montare un nuovo kernel. Il browser funziona anche come gestore per gli aggiornamenti.



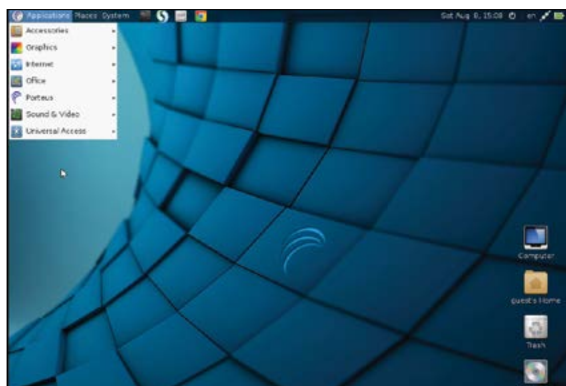
Il verdetto Distro Rolling Release

Le tre distribuzioni che abbiamo messo a confronto in questa pagina sono pensate per evitarvi complicazioni nella costruzione da zero di un sistema. Sia Gentoo sia Arch, su cui si basano Sabayon, Manjaro e Antergos, sono rolling release tra le migliori in circolazione, capaci di distinguersi proprio per la flessibilità nella realizzazione di una piattaforma altamente personalizzata. Sabayon è forse la migliore distro rolling basata su Gentoo. Consente agli utenti meno esperti di confrontarsi con la potenza tipica della distro madre. Il punto forte di Sabayon è proprio **Rigo Application Browser** che si dimostra un meraviglioso front-end grafico per la gestione dei pacchetti di Gentoo. Se invece siete alla ricerca di una rolling release basata su Arch, allora potete scegliere senza esitazioni Manjaro o Antergos. Sono entrambe molto simili e puntano a rendere più accessibile il mondo non proprio semplice di Arch. Le distro, inoltre, hanno installer quasi identici e sfruttano lo stesso gestore di pacchetti grafico. Manjaro, tuttavia, passa avanti ad Antergos sia per quanto riguarda la quantità di applicazioni installate sia per l'uso di Xfce al posto di Gnome 3. Il primo, essendo meno esoso di risorse, consente a Manjaro di essere utilizzato anche su computer dotati di hardware poco potente.



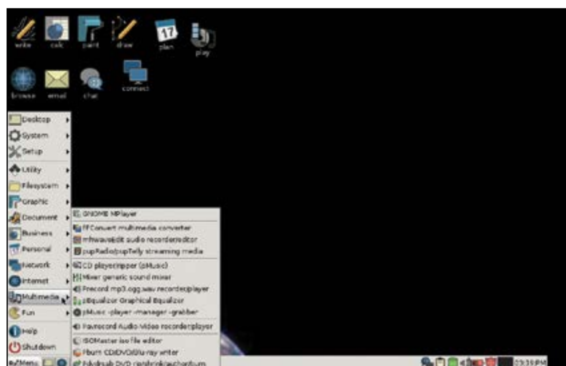
Distro leggere

Per chi vuole computer scattanti



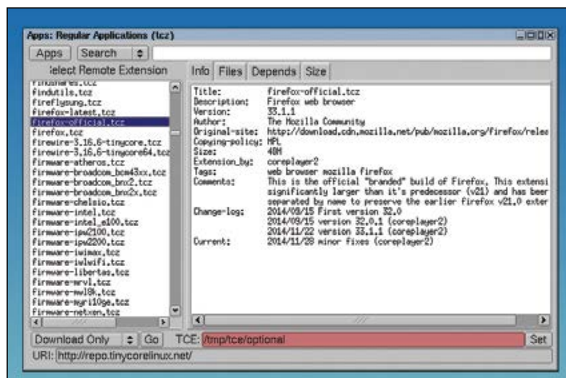
Porteus 3.1

Porteus è una distro unica nel suo genere. Non mette a disposizione un singolo file da scaricare, bensì può essere assemblata tramite una procedura Web. Potete scegliere se dotarla di desktop **KDE 4**, **Mate**, **LXDE** o **Xfce**. Inoltre, siete in grado di sfruttare una vasta serie di software, tra cui **Firefox**, **Chrome** **Opera**, **LibreOffice**, **AbiWord**, **Skype**, driver grafici Nvidia, **AMD Radeon** e via dicendo. Gli utenti più esperti possono anche personalizzare i parametri di boot e attivare i vari moduli del kernel. L'installer funziona anche tramite chiave USB. Come gestore pacchetti, Porteus usa **Slackware**.



Slacko Puppy 5.7

Puppy Linux è una delle distro più leggere in circolazione. Il progetto ha diverse varianti che si differenziano per alcune caratteristiche base. **Wary Puppy** è perfetto per l'uso su PC datati, **Lucid Puppy** è invece costruito sui binari di Ubuntu, mentre **Slacko Puppy** utilizza quelli di Slackware. Slacko ha poi il pregio di sfruttare JWM, un Windows Manager tra i meno ingombranti. La distro, indipendentemente dalla versione, fornisce un'ampia varietà di applicazioni multimediali che oltre a consentire la gestione di questo genere di file, ne permette la creazione. Il browser Firefox è incluso con tutti i tipi di plug-in. In più, è possibile scaricare un componente aggiuntivo che consente di installare **Flash** non presente per impostazione predefinita.



Tiny Core Linux 6.3

Tiny Core è la distro più piccola che riesca a far girare un desktop grafico. Non deriva dalle distribuzioni tradizionali, bensì si basa su un'accurata scelta di componenti estremamente leggeri. Tiny è disponibile in diverse versioni. Oltre a quella che raccomandiamo dal peso di soli 15 MB, c'è anche la variante dotata di sola riga di comando (10 MB). Infine, troviamo la release all-in-one da 72 MB che include diverse funzioni aggiuntive, tra cui il supporto hardware per le reti wireless. Come è lecito aspettarsi, la distro è estremamente veloce. Il Windows Manager predefinito è **FLWM** che si distingue per flessibilità. Tiny Core utilizza poi un proprio formato di pacchetti. Nei suoi repo sono disponibili centinaia di applicazioni popolari come **Firefox**, **LibreOffice**, **Chrome**, **Thunderbird** e molte altre ancora.

Il verdetto Distribuzioni leggere

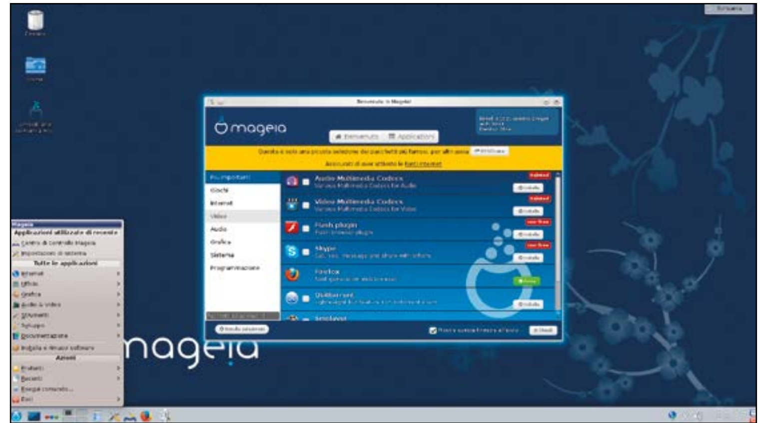
Se avete bisogno di una distribuzione utile a supportare l'hardware più datato, allora dovrete puntare tutto su **Puppy Linux**. Se però volete ridare vita a una vecchia macchina e portarla al passo con le esigenze dei desktop Linux contemporanei, **Tiny Core** è perfetto. L'unica aspetto da considerare se scegliete Tiny è il tempo da dedicare alle applicazioni aggiuntive. La distro, infatti, viene fornita con i soli componenti base. Per renderla pienamente operativa e adatta alle vostre esigenze, dovrete passare diverse ore a metterla a punto. Tenete presente che nonostante la vasta disponibilità di applicazioni, ci vuole tempo per rendere Tiny pienamente operativa e paragonabile a una distro desktop normale. Basta pensare che il primo software che dovrete scaricare è l'installer stesso, il quale non viene fornito nella versione da 15 MB.

La semplicità d'uso non è uno dei cavalli di battaglia neppure per **Slacko**. Oltre a fornire un numero incredibile di strumenti utili per l'uso nei computer più vecchi, mette a disposizione applicazioni personalizzate. Queste, però, possono richiedere più o meno esperienza in base all'uso. A suo vantaggio c'è l'ottima documentazione disponibile sul sito Web, capace di aiutarvi nei vari momenti di bisogno.

Il verdetto

Migliori distro 2015

Chi ama Linux e il software Open Source in generale avrà sempre un vantaggio sugli altri sistemi operativi: la vastità di distribuzioni su cui puntare. Scegliere quella più adatta alle proprie necessità non è mai una cosa facile. In ogni modo cercheremo di indirizzarvi verso quella che, secondo noi, può avere l'impatto più ampio possibile nei vari ambiti di utilizzo. Ubuntu e Fedora, per esempio, sono molto più di semplici distribuzioni. Potremmo invece paragonarle a veri ecosistemi che puntano alla più vasta diffusione: desktop, server e mobile. Se però non amate Unity, potete comunque sfruttare i vantaggi di Ubuntu usando una delle sue tante derivate. Allo stesso modo, se non riuscite ad andare d'accordo con Fedora, si può beneficiare della semplicità di **Korora**. Per quanto riguarda Linux Mint, seppure la distro vanti una grande comunità, in linea di massima il progetto è guidato da un solo individuo. Mint può beneficiare di parecchie donazioni, ma deve comunque fare i conti con i soldi in cassa da destinare alla progettazione. Un problema, questo, che Ubuntu, Fedora, OpenSUSE e Mageia non hanno. Il punto di forza di Mint è sicuramente il desktop Cinnamon che ha fatto da catalizzatore all'ascesa della distro. Cinnamon, tuttavia, non è più una sola prerogativa di Mint, ma può essere utilizzato anche con molte altre distribuzioni, sia come spin ufficiale sia come scelta secondaria. Siamo quindi rimasti con due distribuzioni KDE basate su RPM: OpenSUSE e Mageia. Sul primo non possiamo certo esprimerci in termini negativi, tuttavia rimanere indietro rispetto a Mageia per questioni non prettamente tecniche. Quest'ultima distro, infatti, è considerata come la paladina del movimento Open Source. La sua gestione è democratica e offre un'ampia scelta di ambienti desktop. L'obiettivo, infatti, è rendere la distribuzione utilizzabile da una vasta schiera di utenti che possono contare su strumenti utili alla più ampia personalizzazione. L'ultima release dispone perfino del supporto per UEFI. In definitiva, è proprio Mageia a offrire la migliore combinazione tra semplicità d'uso, flessibilità e praticità.



› *Mageia* **sfrutta sia programmi Open Source sia applicazioni proprietarie**

1° Mageia 5.0 ★★★★★

» *Una distro che ha imparato molto dagli errori commessi in passato e ora vanta grande flessibilità.*

2° OpenSUSE 13.2 ★★★★★

» *Molto pratica da usare. Si distingue soprattutto per la disponibilità di vari ambienti desktop.*

3° Mint 17.2 ★★★★☆

» *Sfrutta al meglio la sua comunità per mettere a disposizione uno dei migliori desktop in circolazione.*

4° Fedora 22 ★★★☆☆

» *Una delle migliori distro Linux, fiore all'occhiello dell'ambiente desktop Gnome.*

5° Ubuntu 15.04 ★★★☆☆

» *Ideale per chi ama Unity e vuole cavalcare l'onda dell'innovazione.*

A voi la parola...

Non siete d'accordo con le nostre scelte? Avreste usato altre distro? Inviate le vostre opinioni su questo confronto a: **recensioni@linuxpro.it**

Considerate anche...

Una rapida occhiata a **http://distrowatch.com** vi fornirà una prima panoramica sull'immensa quantità di distro disponibili. **PCLinuxOS** e **Chakra** sono due popolari release semi-rolling progettate per utenti desktop. Entrambe sfruttano il desktop KDE. Chakra, di solito, è una delle prime piattaforme a implementare le nuove versioni di questo desktop. Se vi trovate bene con Ubuntu, ma non riuscite a digerire Unity, potete usare **Kubuntu** che ha in Gnome il suo ambiente predefinito. Se invece amate KDE, date un'occhiata a **Netrunner**. Se avete un vecchio computer da riportare in vita, prendete in considerazione **Lubuntu** e **Xubuntu** che usano rispettivamente LXDE e Xfce. Mageia e Mandriva hanno poi generato due distro simili. **OpenMandriva** è tra le versioni più vecchie e si concentra esclusivamente su KDE. **ROSA**, invece, è molto più recente e punta tutto su un'innovativa gamma di strumenti. LXP



Crediti

Autori:
Nate Cardozo,
Kurt Opsahl,
Rainey Reitman

Editor:
Parker Higgins, Dave Maass

Presentazione:
Parker Higgins
Pubblicazione della Electronic Frontier Foundation, 2015

Trovate l'originale su:
www.eff.org/who-has-your-back-government-data-requests-2015

Questo articolo è derivato ed elaborato a partire dal report *Who Has Your Back? 2015: Protecting Your Data From Government Requests* di Electronic Freedom Foundation

CHI PROTEGGE I TUOI DATI?

Electronic Frontier Foundation ha pubblicato il suo quinto resoconto annuale sulla privacy e la trasparenza online e spiega le implicazioni per tutti i nostri dati

Le nostre vite sono piene di elementi digitali: dai video condivisi sui social network alle app sui cellulari che geolocalizzano le nostre posizioni, dai dati di login per connetterci alla posta elettronica ai documenti che abbiamo salvato e, naturalmente, la cronologia del nostro browser. Informazioni personali, profonde e persino assurde sono trascritte in pacchetti di dati e viaggiano nelle arterie di fibra ottica della Rete.

Se però le nostre vite si sono integrate nel 21° secolo, la legge non tiene il passo. Persino negli Stati Uniti, considerati all'avanguardia della tecnologia, a oggi, il Congresso non ha ancora aggiornato la legislazione del 1986 "Electronic Communications Privacy Act" per riconoscere che le mail salvate per più di sei mesi si meritano la stessa identica protezione di quelle più recenti. Il Congresso sta anche temporeggiando nell'interrompere la sorveglianza indiscriminata dell'Agenzia per la Sicurezza Nazionale (NSA) sulle comunicazioni online e il Paese attende delle riforme da tempo indispensabili. Sia negli Stati Uniti sia nel Regno Unito il governo sta anche considerando delle proposte che renderebbero obbligatorio dare alle autorità delle backdoor alle tecnologie su cui facciamo affidamento per comunicare digitalmente, il che renderebbe la situazione ancora più difficile. In questo clima, ci rivolgiamo sempre di più alle aziende stesse del settore tecnologico per ottenere le procedure più serrate possibile per difendere i diritti degli utenti. Quali aziende, però, staranno dalla parte degli utenti insistendo sulla trasparenza e rigidi standard legali per quanto riguarda l'accesso del governo ai dati degli utenti? E quali renderanno pubbliche le loro politiche, permettendo al mondo e ai loro utenti di giudicare come difendono il nostro diritto alla privacy? Per quattro anni noi membri dell'Electronic Frontier Foundation, organizzazione internazionale no profit per la difesa dei diritti digitali, abbiamo documentato le pratiche dei più grandi provider e aziende che operano su Internet, giudicando le loro politiche pubbliche e segnalando le migliori. Nel corso dei primi quattro anni, abbiamo visto svilupparsi una trasformazione nelle

> "Se le nostre vite si sono integrate nel 21° secolo, la legge non tiene il passo"

pratiche delle principali aziende nel settore tecnologico. Straordinariamente, i giganti della tecnologia hanno iniziato a pubblicare resoconti annuali delle richieste di dati da parte del governo, promettendo di avvisare gli utenti quando le autorità richiedono accesso ai loro dati, e di richiedere un mandato di perquisizione prima di consegnare i contenuti dell'utente. Le migliori pratiche identificate nei primi resoconti di EFF sono diventati in pochi anni gli standard del settore e siamo fieri del ruolo che le nostre analisi annuali hanno svolto nello spingere le aziende a istituire questi cambiamenti. I tempi però sono cambiati e ora gli utenti si aspettano di più. I criteri utilizzati nel giudicare le aziende nel 2011 erano ambiziose allora, ma sono state quasi universalmente adottate negli anni a seguire. Oggi gli utenti devono aspettarsi che le aziende superino gli standard articolati nel primo resoconto. Devono aspettarsi che aziende come Google, Apple, Facebook e Amazon siano trasparenti sul tipo di contenuto che viene bloccato o censurato in risposta a richieste governative, oltre a svelare quali dati cancellati vengono conservati nel caso al governo servisse accedervi in futuro.
Ci aspettiamo anche che queste aziende prendano una posizione di principio contro le backdoor obbligatorie per le autorità.
Nel quinto rapporto annuale *Who Has Your Back?* abbiamo preso i principi fondamentali delle versioni precedenti e le abbiamo riunite in un'unica categoria: "Buone prassi consolidate". Abbiamo anche elevato le nostre aspettative sulla notifica agli utenti e abbiamo aggiunto nuove categorie per evidenziare altre importanti tematiche relative alla trasparenza e ai diritti degli utenti. Abbiamo alzato gli standard per vedere quali sono le aziende leader.

Criteri di valutazione

A questo scopo, abbiamo usato i cinque criteri che seguono per valutare le pratiche e le politiche delle aziende:

1 Buone prassi consolidate: una categoria composita che valuta le aziende in base a tre criteri a cui devono rispondere:
» L'azienda richiede che il governo ottenga un mandato da un giudice per consegnare il contenuto delle comunicazioni di un utente?
» L'azienda pubblica un *transparency report*, cioè dati regolari e utili relativi a quante volte i governi hanno richiesto informazioni degli utenti all'azienda e quante volte sono state fornite?
» L'azienda pubblica delle linee guida per le autorità spiegando come risponde alle richieste di dati da parte di enti governativi?

2 **Comunicare agli utenti le richieste di dati da parte delle autorità: per ottenere una stella in questa categoria le aziende devono garantire di notificare agli utenti quando il governo richiede i loro dati, tranne in casi in cui sia proibito dalla legge,** in situazioni di emergenza molto specifiche e definite o a meno che farlo non sia futile e privo di efficacia. La notifica dà agli utenti la possibilità di difendersi contro richieste non giustificate di dati da parte del governo. La pratica migliore è avvisare gli utenti prima del passaggio di dati, in modo che possano contrastare l'operazione a livello legale. Abbiamo cambiato questo criterio rispetto agli anni scorsi: quello che richiediamo ora è che l'azienda avvisi gli utenti prima del passaggio di dati tranne in casi in cui sia proibito dalla legge e in situazioni di emergenza e che l'azienda si impegni anche a dare notifica al termine della situazione di emergenza o del periodo di applicazione della legge. Mentre stendevamo i criteri per il nuovo report, l'anno scorso, abbiamo comunicato alle aziende che avremmo introdotto questo cambiamento in modo da dar loro un anno intero per implementare procedure di notifica post factum quando appropriate.



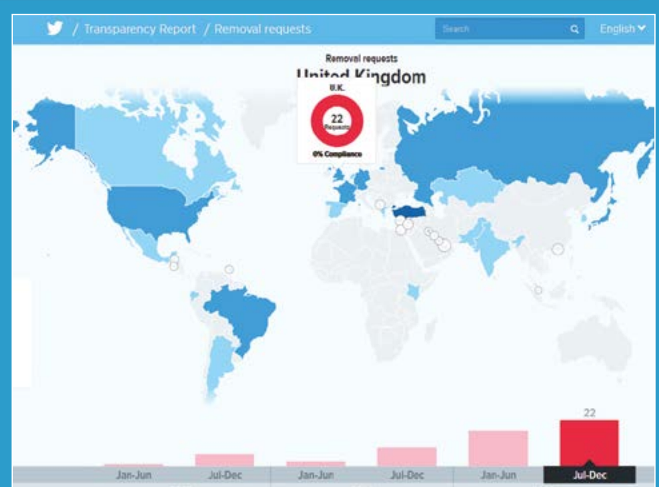
› **EFF ha alzato le aspettative nel 2015**

3 **Rendere pubbliche le politiche di archiviazione dei dati dell'azienda: questa categoria premia le aziende che rendono noto per quanto tempo conservano dati sui loro utenti che non sono accessibili agli utenti stessi (inclusi i log degli indirizzi IP e i contenuti cancellati) in una forma accessibile alle autorità giudiziarie.**
Nel caso il periodo di conservazione dovesse variare per motivi tecnici o di altra natura, l'azienda deve renderlo noto e pubblicare una media approssimativa o un intervallo tipico, insieme a un limite massimo se disponibile.



Richieste di rimozione da parte delle autorità

Per più di un anno, il ricercatore investigativo principale di EFF Dave Maass ha studiato come Facebook coopera con i sistemi carcerari negli Stati Uniti per bloccare l'accesso dei detenuti al social network. Facebook ha persino organizzato uno specifico modulo "Richiesta di eliminazione account detenuto" per aiutare le guardie carcerarie a segnalare la cancellazione degli account di detenuti.
Questa pratica ha ispirato la più recente delle categorie di EFF: controllare con che frequenza le aziende rimuovono contenuti o cancellano account su richiesta delle autorità governative.
Per avere un riscontro positivo in questa categoria, le aziende non hanno bisogno di rifiutare in tutto o in parte le richieste di rimozione dei contenuti da parte delle autorità, ma essere trasparenti sulla frequenza con cui bloccano o rimuovono contenuti o account. Sulle 24 aziende valutate nel nostro report 15 hanno ricevuto una stella in questa categoria, anche se alcune non ospitano contenuti. Un esempio particolarmente brillante di questa pratica sono i dati pubblicati da Twitter, che includono una mappa che consente agli utenti di passare con il mouse sui vari Paesi e ottenere dati sulle richieste di rimozione di contenuti su un periodo di sei mesi.



› *Twitter* offre una panoramica esaustiva di tutte le sue richieste di eliminazione di contenuti e della frequenza con cui vengono accettate

» Abbiamo dato questa stella a tutte le aziende che rendono note le loro pratiche, anche quando queste pratiche sono fortemente scoraggiate dall'EFF, per esempio se l'azienda conserva i dati sugli utenti indefinitamente.

4 Rendere noto quante volte le autorità richiedono la rimozione di contenuti o account degli utenti e con che frequenza l'azienda implementa queste richieste: attualmente è una pratica standard del settore avere transparency report. Riteniamo che la responsabilità di trasparenza delle aziende includa non solo svelare quando le autorità richiedono i dati degli utenti, ma anche la frequenza con cui chiedono la rimozione di dati o la sospensione di account e quante volte l'azienda dà seguito alle loro richieste. Diamo una stella in questa categoria alle aziende che pubblicano regolarmente queste informazioni, nel loro *transparency report* o in un'altra forma similmente accessibile. Le aziende dovrebbero includere i procedimenti legali formali oltre alle richieste informali da parte delle autorità, dato che la censura può prendere diverse vesti.

5 Politiche pubbliche a tutela degli utenti - opporsi alle backdoor: ogni anno, dedichiamo una categoria alla posizione pubblica delle aziende su una certa situazione. Per tre anni, abbiamo premiato le aziende che lavoravano pubblicamente per aggiornare e riformare la legislazione sulla privacy elettronica. L'anno scorso abbiamo dato risalto alle aziende che si opponevano pubblicamente alla sorveglianza di massa. Quest'anno, dato il rinvigorirsi del dibattito sulla crittazione, abbiamo chiesto alle aziende di prendere una posizione pubblica contro l'inclusione forzata di debolezze nella sicurezza o altre forme di backdoor obbligatorie. La posizione può essere presa tramite un post in un blog, nel transparency report, firmando pubblicamente una lettera di protesta o attraverso un altro mezzo pubblico, ufficiale e scritto.
Ci aspettiamo che questa categoria continui a evolversi, per poter tener traccia della posizione delle aziende su un'ampia gamma di questioni legate alla privacy.

Il buono, il brutto e il cattivo

Siamo lieti di annunciare che nove aziende hanno ottenuto stelle in tutte le categorie disponibili (vedi tabella). Bisogna sottolineare che alcune aziende ospitano quantità limitate o irrilevanti di contenuto e di conseguenza la trasparenza sulla richiesta di rimozione di dati da parte delle autorità può non applicarsi. Questi operatori mostrano che è fattibile per le grandi aziende del settore tecnologico adottare delle pratiche d'eccellenza per la trasparenza e tutelare i loro utenti quando le autorità fanno le loro richieste. Sfortunatamente, non tutti stanno applicando queste pratiche.

› I risultati completi del report annuale di EFF evidenziano i risultati molto modesti del popolare servizio di messaging WhatsApp

	L'azienda segue le buone prassi consolidate	Comunica agli utenti le richieste di dati delle autorità	Rende note le politiche di archiviazione dei dati dell'azienda	Rende note le richieste di rimozione di contentuti	Politiche di tutela utenti: si oppone alle backdoor
Adobe	★	★	★	★	★
amazon.com	★	☆	☆	★	★
	★	★	★	★	★
at&t	★	☆	☆	N/A	☆
COMCAST	★	☆	★	N/A	★
CREDO mobile	★	★	★	★	★
Dropbox	★	★	★	★	★
facebook	★	★	★	☆	★
Google	★	☆	☆	★	★
LinkedIn	★	★	★	☆	★
Microsoft	★	★	☆	☆	★
Pinterest	★	★	☆	★	★
reddit	★	★	★	★	☆
slack	★	☆	★	★	★
snapchat	★	☆	★	N/A	★
SONIC.	★	★	★	★	★
tumblr.	★	★	☆	☆	★
	★	☆	★	★	★
verizon	★	☆	☆	★	☆
WhatsApp	☆	☆	☆	N/A	★
WICKR	★	★	★	N/A	★
WIKIMEDIA	★	★	★	★	★
WordPress.com	★	★	★	★	★
YAHOO!	★	★	★	★	★

Due grandi operatori delle telecomunicazioni, Verizon e AT&T, hanno ottenuto risultati particolarmente modesti, perpetuando una tendenza dei grandi provider delle comunicazioni a non tenere il passo con il resto del settore tecnologico già identificata nei precedenti report. Ci sono però anche fornitori di servizi Internet (ISP) e provider di servizi di telecomunicazioni che sono all'avanguardia nell'adottare politiche a tutela dell'utente.
In particolare, Credo e Sonic hanno ricevuto stelle in tutte le categorie. Comcast è appena dietro, con 3 delle 4 stelle possibili. Speriamo che altri operatori delle telecomunicazioni adottino questi standard nei prossimi anni. È risultato anche chiaro che le aziende del settore tecnologico sono unite contro le backdoor forzate dalle autorità. Delle 24 aziende che abbiamo valutato 21 hanno fatto dichiarazioni pubbliche contro le backdoor, che minano la sicurezza e mettono in pericolo la privacy degli utenti. ISP, fornitori di servizi cloud, provider di Webmail e social network sono assolutamente allineati nel rifiutare falle nella sicurezza richieste dal governo.

Buone prassi

Questi standard sono stati sviluppati nei quattro anni di report EFF e comprendono tre dei fattori principali: richiedere un mandato prima di trasferire i dati degli utenti, pubblicare transparency report regolari e pubblicare guide per le autorità. Questi due ultimi elementi aiutano gli utenti a capire con quale frequenza e in quali circostanze le aziende rispondono alle richieste di dati da parte delle autorità, mentre pretendere un mandato assicura che ci siano i presupposti legali per la cessione dei dati. Nel 2011, nessuna azienda ha ricevuto stelle in tutte le categorie. Quest'anno, 23 delle 24 aziende valutate hanno adottato queste prassi. È chiaro che sono profondamente radicate nel settore, ma WhatsApp non è al passo.

Notifica agli utenti

Quest'anno non abbiamo chiesto alle aziende di assicurare semplicemente di informare gli utenti delle richieste di dati da parte delle autorità, ma di avvisarli prima di passare le informazioni. Nei casi in cui le aziende non possono legalmente farlo, abbiamo chiesto che dessero notifica agli utenti non appena consentito dalla legge o al termine dell'emergenza che lo rendeva impossibile. Sapendo che avrebbe richiesto cambiamenti notevoli dal punto di vista tecnico e logistico, abbiamo dato alle aziende un preavviso di oltre un anno dell'inclusione di questo criterio.
Due operatori, Google e Twitter, che avevano precedentemente ottenuto credito nel nostro

Liberi in Linux

Il report EFF è molto centrato sugli Stati Uniti ma, dato che gran parte del mondo sfrutta molti di questi servizi basati negli USA, è un'analisi valida anche internazionalmente. Dato che sono appassionati di Open Source, i nostri lettori sono più consapevoli delle implicazioni della privacy e meglio attrezzati per fronteggiare la situazione. Possono per esempio sfruttare **OwnCloud**, che diventa sempre più efficiente con il passare del tempo, per creare il proprio sistema personale di collaborazione e condivisione di documenti basato sul cloud. Questo significa che si possono mettere in atto delle strategie personali per sfuggire alle regole delle grandi aziende, alle loro normative e ai problemi di privacy. Nella realtà dei fatti però non tutti possono farlo ed è nell'interesse di ognuno di noi che le aziende che forniscono servizi online lo facciano in modo da proteggere gli utenti senza piegarsi ciecamente a ogni richiesta delle autorità. O quanto meno facciano sapere al pubblico come vengono salvati i loro dati e quando (se mai dovesse succedere) vengono condivisi con gli enti governativi. I servizi cloud continuano a crescere, come la quantità dei dati che ci salviamo. Noi continueremo a esaminare per voi le migliori opzioni Open Source e l'emergere di novità come **www.onlyoffice.com**.



› Creare i tuoi servizi cloud indipendenti usando *OwnCloud*, è un modo per tutelare la tua privacy

report per aver avvisato gli utenti delle richieste di dati da parte delle autorità, quest'anno non hanno ricevuto stelle perché non avevano in atto procedure per avvisare gli utenti quando fosse tornato possibile per legge o cessata l'emergenza. Delle 24 aziende valutate, 15 soddisfacevano questo criterio e ci fa piacere vedere che il settore si sta muovendo in questo senso. Ci ha colpito particolarmente, in positivo, la policy di Dropbox, che afferma: "La politica di Dropbox è di notificare gli utenti nel caso le autorità richiedano i loro dati prima di consegnarli, salvo proibizione a norma di legge. L'avviso potrebbe essere inoltrato in ritardo nei casi che includano minacce fisiche o di morte oppure lo sfruttamento di minori".

Conservazione dei dati

Quest'anno per la prima volta abbiamo valutato le aziende anche per la trasparenza su quali dati cancellati continuano a conservare. Spesso gli utenti non si rendono conto che i dati che cancellano da un provider di posta elettronica o un social network rimangono salvati e disponibili per le autorità giudiziarie su richiesta. La trasparenza è il primo passo per far comprendere agli utenti cosa succede ai loro dati cancellati, quindi valutiamo le aziende sotto questo aspetto. Va specificato che non facciamo richieste specifiche sul fatto che gli operatori cancellino i dati dopo un determinato periodo di tempo. In realtà, alcune aziende affermano pubblicamente di conservare i dati cancellati e log dei server indefinitamente, una pratica che secondo noi è terribile per gli utenti. D'altra parte, per questo report, chiediamo solo agli operatori di essere chiari sul periodo di archiviazione di dati che non risultano facilmente visibili per l'utente (inclusi indirizzi IP e dati DHCP) oltre che dei contenuti che gli utenti hanno cancellato. Anche in questo caso, 15 aziende su 24 hanno risposto a questo criterio. Siamo stati particolarmente colpiti dalla chiarezza e dal dettaglio dei termini di Comcast. L'azienda mantiene i dati sui dettagli delle chiamate per il servizio telefonico Xfinity Voice per due anni. Includono chiamate locali, locali a pagamento e su lunga distanza. In casi particolari, potrebbero essere disponibili anche dati più vecchi ma è necessario impiegare più tempo e risorse per reperirli. Per ottenere più dettagli su questa esemplare descrizione della politica di conservazione dei dati leggete il **Comcast Law Enforcement Handbook** all'indirizzo **http://bit.ly/LXFitsthelaw**.

Opporsi alle backdoor

Uno dei grandi trend che osserviamo nel settore è il rifiuto delle debolezze nella sicurezza richieste dal governo. 21 delle 24 aziende considerate hanno preso una posizione pubblica contro l'uso delle backdoor. Si tratta di uno schieramento chiaro di cui le autorità dovrebbero tenere conto a livello legislativo. Molte aziende hanno sottoscritto una lettera organizzata dall'Open Technology Institute contro le richieste di abbassare intenzionalmente la sicurezza, che dichiara: "Vi esortiamo a respingere qualsiasi proposta di abbassare deliberatamente i livelli di sicurezza dei nostri prodotti… che le si chiami *front door* o *back door*, introdurre intenzionalmente delle vulnerabilità in prodotti sicuri a beneficio dell'uso da parte delle autorità rende i prodotti in questione meno sicuri anche contro attacchi di altra natura. Tutti gli esperti di sicurezza informatica che si sono espressi pubblicamente sull'argomento sono d'accordo su questo punto, inclusi gli esperti del governo degli USA".

Le conclusioni di EFF

Siamo lieti di vedere che le principali aziende del settore tecnologico competono sul piano della tutela della privacy e dei diritti degli utenti. Pratiche che incoraggiano la trasparenza con gli utenti sulle richieste di dati da parte delle autorità stanno diventando la norma per le aziende che operano su Internet. Anche se siamo in grado di giudicare solo una piccola parte del settore tecnologico, crediamo che la nostra analisi rappresenti un più ampio spettro. Forse stimolate dai dibattiti sulla sorveglianza governativa e in risposta alla crescente attenzione pubblica a questi aspetti, sempre più aziende stanno volontariamente esprimendo il desiderio di limitare le richieste di accesso ai dati da parte delle autorità e di dare agli utenti i mezzi per opporvicisi. Pensiamo che questo tipo di trasparenza possa portare sia a una discussione più ampia delle tematiche sia a cambiamenti estesi su come e quando i governi possano accedere ai dati degli utenti, e favorire il miglioramento e l'estensione delle leggi sulla privacy digitale. Ci rendiamo anche conto del fatto che le aziende del settore tecnologico sono nella posizione di conoscere e contrastare le richieste troppo estese delle autorità, quindi dobbiamo fare quanto in nostro potere per incoraggiarli a rendere pubbliche le loro conoscenze e a opporsi.
Nel consegnare i nostri dati a queste aziende, abbiamo dato loro la grande responsabilità di fare tutto il possibile per tutelare la nostra privacy. Siamo felici di constatare che molte delle aziende valutate hanno saputo fronteggiare questa sfida. LXP

Lambda Lady

Un'intervista in esclusiva con Katie Miller per discutere la programmazione funzionale, OpenShift e come far interessare più donne alla programmazione

Katie Miller è una OpenShift Developer Advocate e programmatrice funzionale esperta in diversi linguaggi. È stata coautrice del libro *Getting Started with OpenShift* e tiene numerose conferenze sulla demistificazione della programmazione funzionale. Katie è anche la co-fondatrice del gruppo Lambda Ladies (www.lambdaladies.com), una comunità online per donne interessate alla programmazione funzionale. Dopo aver lavorato per Red Hat come ingegnere del software e Developer Advocate (ossia uno specialista che si occupa di rendere una piattaforma facile da usare per gli sviluppatori), Katie ha recentemente iniziato a lavorare con Facebook.

Linux Pro: **Il tuo background in origine era di tipo giornalistico, poi sei passata alla programmazione: quale delle due carriere secondo te può offrire delle prospettive migliori?**
Katie Miller: Oh, per chi? Per la gente che sta finendo gli studi in questo periodo? Direi che il giornalismo è interessante in questo momento. Il settore dei mezzi di comunicazione è in uno stato di continua trasformazione secondo me. Sarebbe difficile definire come sarà nel corso dei prossimi anni. Ci potrebbero essere delle ottime opportunità per persone adattabili e ricche di idee. D'altra parte, però, anche il mondo della tecnologia è molto dinamico ed entusiasmante e i mezzi di comunicazione stanno ovviamente sfruttando sempre di più la tecnologia, quindi ci sono dei punti in comune tra i due settori.

LXP: **I tuoi studi universitari sono stati nell'ambito del giornalismo o della programmazione?**
KM: Sono laureata in giornalismo e ho fatto un periodo di collaborazione con un quotidiano. Sono una giornalista professionista a tutti gli effetti. Dopo alcuni anni di giornalismo sono tornata agli studi e ho conseguito un master in informatica specializzandomi in ingegneria del software. È stata proprio questa qualifica ad aprirmi le porte per il primo lavoro in questo settore.

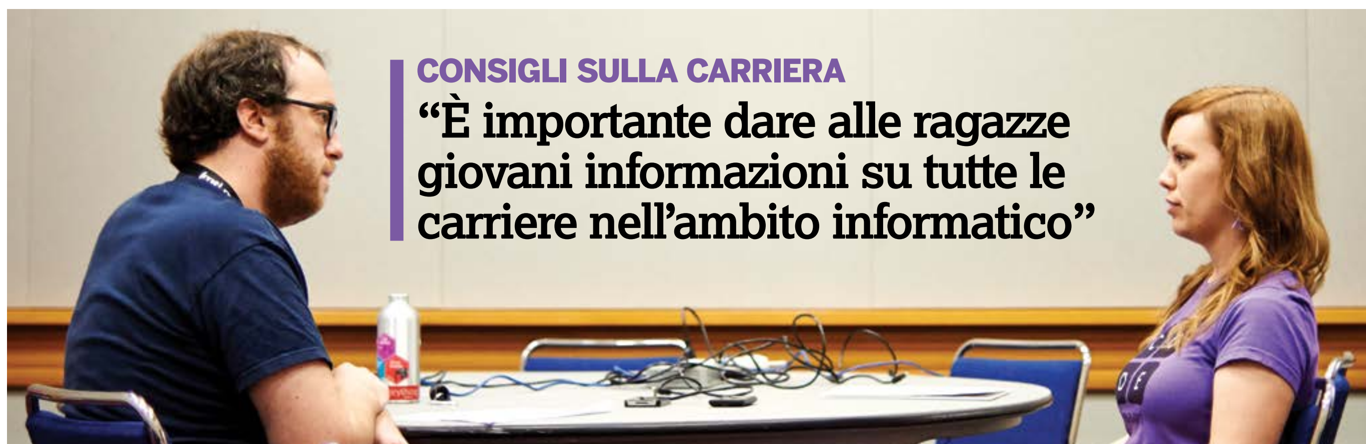
LXP: **Quali sono state le motivazioni che ti hanno spinta a passare dal giornalismo alla tecnologia? C'è stato un momento specifico in cui hai deciso che volevi passare da una carriera all'altra?**
KM: Per capire come è avvenuto questo passaggio dovresti conoscere un po' della mia storia personale. Sono cresciuta a stretto contatto con la tecnologia: i miei due fratelli maggiori sono programmatori e quando ero una ragazzina mi hanno insegnato QBasic e cose del genere. Mi ricordo che da teenager mi divertivo a sperimentare con il Basic e a creare i miei siti, rubando pezzettini di JavaScript qua e là (ride) e cercando di farli funzionare! Quindi ho avuto familiarità con il codice di programmazione per tutta la mia infanzia, non era un argomento che non conoscevo. Dopo qualche anno di giornalismo, ho ricominciato lentamente a riavvicinarmi a quel settore: ho lavorato diversi anni come reporter, poi ho iniziato a lavorare come redattore aggiunto responsabile del controllo qualità. Poi ho cominciato a collaborare con quotidiani e siti Web, quindi stavo in un certo senso avvicinandomi sempre più alla tecnologia. A un certo punto mi hanno chiesto di lavorare con HTML, CSS e un po' di PHP: mi è venuto molto naturale, perché avevo già quel tipo di background a differenza di molti dei colleghi che avevo al mio fianco. Mi sono resa conto di quanto fosse divertente per me e ho pensato che, dato lo stato di una serie di aspetti nel mondo della comunicazione giornalistica e dei vari mutamenti in atto, fosse il momento opportuno per approfondire le mie



conoscenze di programmazione. Una volta tornata all'università per studiare informatica mi sono reinnamorata completamente del codice e mi sono resa conto che era quella l'attività che avrei voluto svolgere a tempo pieno.

LXP: **Nel Regno Unito, e in Europa in generale, c'è la percezione che il sistema universitario non faccia abbastanza per incoraggiare le donne a perseguire corsi di studio in ambito tecnologico. Hai incontrato qualche difficoltà legata al fatto che sei una donna?**
KM: Intendi che mi abbiano scoraggiata direttamente o che non mi abbiano dato abbastanza incoraggiamento?

LXP: **Intendo una mancanza di incoraggiamento che però per una ragazzina giovane può avere effetti negativi.**
KM: Beh, in questo caso direi che è un problema globale, certamente non relativo »



CONSIGLI SULLA CARRIERA

"È importante dare alle ragazze giovani informazioni su tutte le carriere nell'ambito informatico"



» solo al Regno Unito. Per quanto mi riguarda, crescendo in una famiglia con dei fratelli maggiori che si occupavano di programmazione, mi sono trovata in un ambiente ideale: non tutti hanno questo tipo di esposizione all'informatica. Avevo anche studiato informatica tra le materie del liceo. Alla fine delle superiori, dovendo scegliere una carriera, avevo sia l'opzione del giornalismo, dato che mi piace scrivere, sia quella degli studi tecnologici. Quando ci ripenso, mi piacerebbe provare a ricercare alcune delle cose che mi sono appuntata in quel periodo per avere una conferma, ma secondo me non sapevo neanche la linguistica computazionale fosse un'opzione. Guardando indietro mi domando perché non ci abbia pensato: sarebbe stato perfetto per me cercare il punto di incontro tra linguaggio naturale e informatica. Il punto per cui mi interessava la scrittura era l'aspetto linguistico più che quello giornalistico ma ciò nonostante non sapevo dell'esistenza di questa opzione di studi perfetta per me. Forse nel mio caso, quindi, è mancata la guida giusta: se avessi avuto qualcuno a darmi dei consigli sul mio percorso di studi e mi avesse fatto notare che non avevo valutato tutte le opzioni probabilmente avrei seguito la strada della Linguistica computazionale. Credo che sia importante avere più informazioni possibile e dare alle ragazze (e ai ragazzi!) giovani una panoramica completa di tutte le possibilità che hanno a disposizione nell'ambito dell'informatica e della tecnologia. Non è detto che si possa diventare solo "programmatori" o "ingegneri del software": ci sono molte altre opzioni anche se spesso non vengono messe in sufficiente risalto. Per questo cerco di essere coinvolta in progetti aperti agli studenti, come i workshop agli Excite Camps, che organizza IBM. Ho fatto un po' di workshop su OpenShift e la programmazione Open Source... ho lavorato anche a un libro per una campagna che stiamo facendo in Australia dal nome **Tech Girls are Superheroes**, ossia le ragazze tecnologizzate sono supereroi, come parte del movimento **Tech Girls**. Il libro viene distribuito gratuitamente alle ragazze delle superiori e delle medie in tutto il Paese per dare loro dei modelli di riferimento per quanto riguarda le donne che lavorano nella tecnologia.

LXP: Come è iniziato il tuo coinvolgimento con Tech Girls are Superheroes? Ti hanno contattata loro o ti sei offerta spontaneamente?
KM: Conoscevo già Janine, la fondatrice del progetto, e mi ha proposto di scrivere una storia. È interessante il modo in cui è strutturato il libro. Scrivi una storia in prima persona nei panni di un personaggio che ha un alter ego supereroe e sviluppi una narrativa a partire da questo presupposto. È uno schema originale! Non scrivi di te stesso, il personaggio non ti ritrae specificamente, ma potrebbe avere alcune delle tue caratteristiche. Così il mio personaggio si chiama Dixi e il suo superpotere è di poter scrivere e parlare in qualsiasi linguaggio, naturale o informatico che sia.

ISPIRARE LE NUOVE GENERAZIONI
"Puoi lavorare a progetti che entrano nelle vite di milioni o persino miliardi di persone"

LXP: Questa caratteristica è basata sulla tua vita reale? Avevi un interesse nella linguistica naturale e questo ti ha portato ai linguaggi di programmazione?
KM: Sì, sono interessata ai linguaggi in generale. Vedo dei paralleli tra quelli naturali e informatici. Quando la gente scopre che ero una giornalista e poi ho iniziato a lavorare con l'informatica pensano che sia strano ma a me non sembra. Come redattore, devi essere precisissimo: ogni lettera ha la sua importanza. Puoi cambiare l'intero significato di una frase con una lettera. Succede la stessa cosa con il codice di programmazione. Ti dimentichi un punto e virgola o qualcosa di altrettanto apparentemente secondario e non funziona più nulla. Ci sono quindi caratteristiche importanti in entrambe le professioni.

LXP: Anche la ricerca delle radici etimologiche, cercare di capire come certi linguaggi si siano evoluti e differenziati l'uno dall'altro è simile allo studio dei linguaggi di programmazione.
KM: Sì, penso che ci siano molti punti in comune.

LXP: Nel sistema didattico del Regno Unito uno degli obiettivi è che il governo cerchi di introdurre la programmazione per i ragazzi più giovani. Ti sembra un obiettivo importante? Come si potrebbe sfruttare per incoraggiare più donne a entrare nel settore informatico?
KM: Sicuramente. Sono entusiasta dei programmi del Regno Unito e so che l'Estonia ha già fatto dei passi in quella direzione. Mi piacerebbe che si procedesse in questo senso anche in Italia e in altri Paesi. Ci sono tante iniziative gestite da volontari che cercano di esporre i bambini alla programmazione e questo è fantastico ma l'unico modo per farlo davvero su larga scala è introdurre l'informatica nelle scuole. Penso assolutamente che sia un provvedimento importante, in particolare per le ragazze che, come sottolineavi, a volte non vengono raggiunte. Secondo me sarebbe utilissimo. Come sai, penso che abbiamo bisogno di più donne nel settore! Ci servono tantissime persone di tipologie diverse per costruire la tecnologia del futuro e renderla veramente accessibile a chiunque. È quindi molto importante assicurarsi che tutti abbiano delle conoscenze di base di programmazione. Naturalmente non tutti diventeranno ingegneri informatici, ma credo che l'alfabetizzazione informatica oggi sia essenziale: non dobbiamo solo insegnare alla gente a usare la tecnologia, ma anche permettere a tutti di gestirla e crearne il futuro.

LXP: È molto importante. Crescendo sviluppi una forte dipendenza dalla tecnologia che utilizzi, ma devi anche sapere come funziona. Se c'è una qualche azienda closed source che decide di cambiare i termini di licenza del suo software e ti ritrovi a condividere più informazioni di quanto non vorresti, senza delle conoscenze di informatica non puoi fare nulla.

KM: Esatto. C'è la percezione che le nuove generazioni siano composte di maghi della tecnologia ma secondo me non è sempre così. Mi sembra proprio che siano solo bravi a sfruttare la tecnologia, senza necessariamente capirla in modo approfondito. Faccio workshop per le ragazze e, per esempio, ne tengo uno sulle app per Android. Le ragazze che hanno gli iPhone non sanno neanche che le app per Android si possono usare sui loro telefoni. Mi stupisce che non abbiano questo tipo di conoscenza. Sono delle utenti efficienti ma non hanno la comprensione dello strumento che deriva dal sistema educativo e permette di raggiungere un livello di conoscenza e sfruttamento ottimale dei propri dispositivi, per poter creare ciò che vorrebbero usare e non solo sfruttare quel che già c'è.

LXP: Qquando ho studiato informatica, le lezioni erano del tipo "Ecco Microsoft Word. Per creare un documento dovete seguire questa procedura, e così via". È incoraggiante oggi vedere il centro dell'attenzione spostarsi da "Ecco come usare Word" a "Come creare il proprio word processor".

KM: Esatto, è importante non essere limitati dall'interfaccia utente che ci si trova di fronte. La sensazione che si prova quando si impara a programmare è di avere il controllo dello strumento. Puoi prendere frammenti di codice da varie fonti e creare tutto ciò che riesci a immaginare... credo che questa possibilità dovrebbe essere accessibile a tutti.

LXP: L'obiettivo del movimento Tech Girls are Superheroes è quindi di mettere in evidenza un superpotere che tutti possono acquisire?

KM: Esatto, e non si parla solo di programmazione. Ci sono numerosi superpoteri diversi che le varie donne hanno, in diversi ambiti. Non servono solo programmatrici ma anche analiste commerciali che hanno una profonda conoscenza del mondo del business e possano fare da tramite tra i tecnici e i commerciali. C'è bisogno di brave project manager, di architetti e amministratrici di sistemi... ci sono tantissime categorie diverse di professioni. Il libro cerca di riflettere su questa esigenza e i diversi modi in cui puoi fare qualcosa di significativo. Credo che la possibilità di "fare la differenza" sia veramente importante per le ragazze e la tecnologia ti dà questa opportunità. Questo è uno dei motivi per cui è un settore entusiasmante: puoi lavorare a progetti che entrano nelle vite di milioni o persino miliardi di persone. Di conseguenza è importante far arrivare il messaggio alle ragazze, farle entrare in contatto con delle persone che possano diventare modelli da emulare, anziché perpetuare l'immagine del programmatore come nerd che lavora in uno scantinato e non ha una vita sociale... o qualsiasi sia lo stereotipo del momento. Bisogna andare oltre.

LXP: Viene spontaneo pensare che le donne possano trovare un ambiente molto favorevole nella comunità dell'Open Source.

KM: Sì, è interessante perché dal punto di vista statistico i numeri sono bassi. Un recente sondaggio di FOSS ha mostrato un lieve miglioramento, ma il precedente dava un 2% di donne nell'ambiente. Mi domando perché... credo che il punto sia l'interazione nella comunità, a volte queste situazioni hanno poco a che fare con la tecnologia... sono argomenti difficili.

LXP: L'errata percezione sociale ha un effetto negativo sia sulle donne sia sugli uomini: lo stereotipo del programmatore nerd e solitario è incredibilmente dannoso.

KM: Certo, non sono solo le donne a venire scoraggiate da queste cose: influenzano moltissime persone diverse appartenenti a varie minoranze. Spesso si parla di differenziare l'ambiente della programmazione e la gente dà per scontato che si parli della carenza di donne ma naturalmente ci sono molti altri aspetti della problematica. Ci sono molti gruppi che non sono rappresentati sufficientemente nella tecnologia.

LXP: In che modo hai iniziato a interessarti a Linux e all'Open Source?

KM: La prima volta che ho usato Linux, e di conseguenza sono entrata in contatto con il mondo dell'Open Source, è stato all'università. Ci è voluto un pochino di tempo perché arrivassi a utilizzarlo anche a casa costantemente. Il primo lavoro che ho svolto era in una banca: avevano un approccio più chiuso naturalmente di quello universitario ma erano molto aperti in relazione all'Open Source, così ho potuto usare molto codice Open Source al lavoro. Programmavo in Java e questo è stato positivo. Alla fine sono arrivata al punto in cui non potevo far fronte alle mie esigenze: non riuscivo a far funzionare correttamente i programmi che mi servivano sul mio laptop. Così ho provato un po' di kernel Linux e alla fine ho creato il mio primo kernel per Fedora (prima di iniziare a lavorare per Red Hat). Ho scoperto che soltanto Fedora funzionava al meglio con il mio hardware specifico, così mi sono creata un bel kernel per Fedora e sono andata avanti da lì. Alla fine, dopo un periodo relativamente breve, sono andata a lavorare per Red Hat! Direi che è stato un bel percorso. Sì, usavo Fedora, lo sfrutto anche adesso sia a livello personale sia al lavoro.

LXP: Il fatto che tu avessi sviluppato un kernel per Fedora ti è stato utile per farti assumere da Red Hat?

KM: Mmm... non lo so ma penso che sia stato un fattore positivo. Immagino che gli abbia fatto piacere, ma è stata una pura coincidenza: Fedora era semplicemente la soluzione ideale che avevo trovato per le mie esigenze.

LXP: Quindi come sei arrivata a lavorare per Red Hat?

KM: Avevo sentito che stavano cercando un ingegnere del software e mi sono proposta. Ho iniziato a lavorare nella loro area di ingegneria e contenuti e sostanzialmente scrivevo software per aiutare i loro autori di contenuti a far bene e facilmente il loro lavoro. Inoltre ero l'esperto di riferimento, rispondevo alle loro domande tecniche.

LXP: Quali sono le responsabilità legate al tuo ruolo?

KM: Sono un Developer Advocate per OpenShift, che è la piattaforma Open Source di servizio di Red Hat. Ho responsabilità come andare a manifestazioni e convention, fare »





» conferenze, scrivere post sul blog, sostanzialmente interagire con gli sviluppatori in tutti i modi in cui posso e aiutarli a capire OpenShift. Do anche loro l'opportunità di fornire un feedback al nostro team di sviluppo, un aspetto che secondo me è molto importante. Essere un Advocate non significa solo comunicare ma anche ascoltare.

LXP: In qualche modo torni alle tue origini come giornalista e alla passione per il linguaggio, facendo quasi da interprete tra le persone che usano il codice e quelle che lo creano...
KM: Sì, a volte la comunicazione tra i due gruppi può non essere del tutto lineare. Noi facciamo da ponte tra queste due categorie, oltre che tra altri gruppi.

LXP: Potresti spiegare ai nostri lettori cos'è OpenShift, cosa fa e in che modo potrebbero trovarlo utile?
KM: OpenShift è utile per due tipologie di persone. Per gli sviluppatori è fantastico perché permette loro di concentrarsi solo sul proprio codice, con *Platform as a Service* (PaaS) siamo a un livello di astrazione sopra l'infrastruttura di servizio. OpenStack, o EC2, sono già ben affermati e ora si parla di PaaS, ambienti come OpenShift, che gestiscono non solo le stesse cose dei server di servizio, ma anche il sistema operativo sul server, il run time, i database e l'upgrade di questi elementi quando ci sono, per esempio, aggiornamenti di sicurezza. Il programmatore non si deve più occupare di tutti questi elementi, quindi deve solo scrivere il suo codice e pubblicarlo. Così ti trovi a lavorare in un sistema che funziona, devi solo inserirci il codice ed è subito online in Internet e pronto all'uso.
È pratico anche per gli amministratori di sistema, perché rende autonomi gli sviluppatori. OpenShift è Open Source, come progetto OpenShift Origin, ma c'è anche una versione per le aziende, OpenShift Enterprise, quindi il sysadmin può scegliere la modalità che gli è più congeniale. Invece di dover dare agli sviluppatori gli stack da usare, per esempio una certa versione di Ruby, di Java e quant'altro, si può dare loro accesso root a un server e possono gestirsi da soli per ideare e creare tutte le magnifiche applicazioni che possono concepire e pubblicarle online senza bisogno dell'intervento dell'amministratore di sistema. Direi che è molto pratico per tutti!

LXP: Il che è sempre un'ottima cosa... gli sviluppatori e gli amministratori di sistema lo hanno apprezzato? Sta andando bene?
KM: Direi proprio di sì! Abbiamo oltre 1,6 milioni di applicazioni su **www.openshift.com**. Devo anche specificare che OpenShift è tre cose diverse. C'è **OpenShift Origin**, che è il progetto Open Source. È su GitHub, con licenza **Apache 2**, quindi puoi prenderlo e usare la tua piattaforma cloud. Poi c'è la versione per le aziende, e abbiamo dei grandi clienti che la usano. Poi c'è anche **OpenShift Online**, una PaS (Platform as Service) pubblica. Anche in questo caso abbiamo un livello *free* molto generoso. Ottieni tre spazi, che noi chiamiamo *gears containers*, ciascuno con un gigabyte di salvataggio e 512 MB di RAM, sufficienti a fare un bel po' di cose. Se hai bisogno di ancora più risorse ci sono dei piani commerciali specifici. Stiamo ottenendo delle grandi soddisfazioni su questo fronte. Lo spazio aziendale ha una grande trazione e sta andando davvero molto bene. Il progetto Open, d'altra parte, è stato tra l'altro nella classifica di GitHub, il *GitHub State of the Octoverse report*, si è qualificato tra i primi cinque prodotti per le *merge pull request*, quindi possiamo dire che è al centro di una fantastica comunità. Uno dei merge più interessanti dello scorso anno è stata un'azienda di nome Uhuru che si è unita nel supporto di .net support, il che è interessante.

LXP: È bello veder nascere nuove realtà, magari con idee innovative e la voglia di provare soluzioni a cui nessuno ha mai pensato prima.
KM: Sì, e dai alla gente nuove alternative da esplorare, no? Il fatto che tu possa sviluppare prodotti .net nativamente è molto positivo ed è una soluzione che potrebbe interessare moltissimi sviluppatori.

LXP: So che tu ti occupi anche di programmazione funzionale. Potresti spiegarci che cos'è Haskell?
KM: Haskell è un linguaggio di pura programmazione funzionale (*functional programming* o FP) e mi piace davvero molto. Il motivo per cui lo apprezzo tanto non è necessariamente il fatto che io pensi che tutti dovrebbero usarlo per ogni tipo di programma... penso però che sia un ottimo linguaggio per imparare i concetti della programmazione funzionale perché è puro ed esteticamente ineccepibile. Se il tuo programma arriva a termine della compilazione è molto probabile che sia corretto. Questo ti permette di evitare un'ampia serie di bug. È uno strumento fantastico per insegnare e anche Simon Peyton Jones, esperto di informatica di fama internazionali e uno tra i principali autori di Haskell, consiglia di imparare questo linguaggio anche se non si intende usarlo perché ti insegna un modo di pensare corretto. Secondo lui studiare Haskell ti fa diventare un programmatore migliore in qualsiasi linguaggio tu decida di utilizzare il tuo codice. Il punto quindi non è che Haskell conquisti il mondo, ma che aiuti a imparare dei concetti fondamentali. Qualunque sia la soluzione che preferisci, per alcuni può essere Haskell, o **elixir** (**http://elixir-lang.org**) un nuovo linguaggio davvero interessante per chi vuole imparare la programmazione funzionale. Ha una sintassi analoga a quella di Ruby, gira sulle macchine virtuali Erlang VMs così puoi sfruttare tutte le magnifiche possibilità di programmazione concorrente che vi si collegano, ed è anche profondamente influenzato da **Clojure** (**http://clojure.org**). Queste caratteristiche unite funzionano molto bene e per chi ha un background di programmazione in Ruby o JavaScript può essere un'ottima soluzione per imparare questi concetti. Naturalmente ci sono anche Clojure, Scala e altre opzioni. Haskell è solo una di esse, ma mi piace in modo particolare, probabilmente perché ho imparato così la programmazione funzionale! Secondo me è un ottimo linguaggio anche perché hai un feedback immediato. Quando cerchi di compilare il tuo programma e non si compila, devi seguire un processo di analisi per sistemarlo e compilare. Quando affronti questo tipo di processo impari davvero molto! [ride].

LXP: Quali sono i benefici che si ricavano dall'uso dalla programmazione funzionale?
KM: Secondo me il beneficio principale della programmazione funzionale è che ti dà la capacità di ragionare sul tuo codice. Il punto è che scrivi il codice usando funzioni di tipo matematico. Sono funzioni che portano solo da input ad output e non fanno nient'altro.

Sono funzioni "senza effetti collaterali". Di conseguenza quando lavori con questo stile produci codice che è davvero molto modulare. Hai tanti piccoli stralci di codice che poi unisci per ottenere i tuoi scopi ed è molto più facile ragionare in termini di tanti microelementi separati, perché non hanno influenza su altre parti del codice (non hanno effetti collaterali…), sono molto facili da testare ed è semplice, per esempio, farli funzionare in parallelo. È un grande beneficio, anche solo in termini di ragionamento, e il modo in cui il codice funziona influenza come lavori, nello sfruttare le funzioni in parallelo e nella fase di testing. Per esempio c'è uno strumento, che si chiama **QuickCheck,** che è portato in moltissimi linguaggi ed è in Haskell, che può automaticamente sottoporre a test un'ampia serie di proprietà del tuo codice, il che è veramente comodo… permette di risparmiare tempo su larga scala! In questo modo sostanzialmente si elimina un'intera classe di bug. Puoi capire moltissimo del tuo codice già solo dalle proprietà in **QuickCheck** e dalla *type signature* (una linea che indica la tipologia di una variabile) in Haskell. Ci sono quindi moltI aspetti che semplicemente non possono essere sbagliati in questo contesto. Naturalmente, ci possono sempre essere bug nel codice, ma ci si libera di un'intera classe di problemi. Haskell semplifica notevolmente la vita del programmatore… io dico sempre che preferisco litigare con un compilatore che dover discutere con qualcuno del motivo per cui alcuni bug sono rimasti nella versione finale di un programma! [Ride].

LXP: Capisco perfettamente il tuo punto di vista! C'è una comunità che si fa sentire molto chiaramente ed è sempre pronta a farti sapere se ha trovato un bug! Per quanto riguarda Haskell, in particolare, sembra che ci siano delle opinioni molto forti. Sai perché? Secondo te perché infastidisce alcuni programmatori?
KM: Secondo me il punto è che Haskell è molto diverso dalla maggior parte degli altri linguaggi, quindi le persone o lo amano o lo odiano. Il linguaggio inoltre è stato progettato da un comitato di persone di grande successo e forse per questo motivo alcuni possono avere la percezione che sia di natura accademica. Secondo me non è così: si sviluppa da origini accademiche, ma non è accurato definirlo accademico. Penso che sia un linguaggio molto pratico anche per il codice di produzione e conosco moltissime persone che lo usano per lo sviluppo commerciale, in particolare nel settore finanziario. In quel segmento è sfruttatissimo. Persino Facebook ora ha il progetto Haskell sviluppato come Open Source usando Haskell. Quindi possiamo sicuramente affermare che grandi aziende lo usino. Non è un linguaggio astratto che vive nella sua torre d'avorio, viene sfruttato in molti contesti pratici. Non saprei davvero dirti perché alcune persone si oppongano veementemente al suo utilizzo, forse è solo… il fatto che è molto succinto, ha una sintassi essenziale e ad alcuni questo approccio non piace. Quando si incontra qualcosa di nuovo e diverso c'è sempre una curva di apprendimento per acquisire una certa dimestichezza con i concetti. Forse non persisti abbastanza da superare questa fase: ti sembra solo un linguaggio strano e diverso dagli altri.

LXP: Forse alcuni programmatori hanno abitudini acquisite e consolidate e pensano che il loro approccio alla sintassi sia l'unico adatto al loro lavoro. In questo caso potrebbero pensare che Haskell sia un tentativo di rivoluzionare il loro mondo.
KM: Certo, naturalmente ci sono anche persone che non vogliono iniziare a lavorare in un nuovo tipo di linguaggio. In quel caso Haskell non è proprio l'ideale!

LXP: Secondo te OpenShift è una tecnologia chiave per Red Hat?
KM: Sì, penso di sì. Red Hat opera su molti fronti diversi attualmente. Quando dici 'Red Hat' alcune persone ancora pensano a Linux per le aziende ma secondo me siamo cresciuti ben oltre questa definizione. Abbiamo molti progetti attivi nel cloud, nel salvataggio dei dati e nell'area dei sistemi operativi e secondo me OpenShift ha un peso rilevante in questo contesto.

LXP: Secondo te è più importante per l'area cloud o nel settore dei sistemi operativi per Red Hat?
KM: Beh, secondo me ci sono delle sovrapposizioni perché gira con **Red Hat Enterprise Linux** (RHEL) e nel nostro settore di sistemi operativi abbiamo iniziative come **Project Atomic**, che è un sistema operativo leggero da usare con contenitori come quelli di Docker. Ci stiamo muovendo anche verso un modello come quello di Docker, quindi ci saranno delle intersezioni tra le aree.

LXP: C'è stata molta comunicazione tra Red Hat e Docker sull'uso di contenitori simili?
KM: Abbiamo annunciato un po' di tempo fa una partnership ufficiale con Docker, quindi stiamo tutti lavorando attivamente insieme per la prossima grande versione di OpenShift, dato che sarà costruita sulla base di Docker. L'architettura di conseguenza subirà delle modifiche importanti, il che è molto interessante e coinvolgente! Docker è diventato lo standard per i contenitori, quindi ha senso muoversi in questa direzione.

LXP: Quindi sei d'accordo con Docker che la virtualizzazione tramite container, e non il modello HyperVisor, sia la soluzione migliore?
KM: Beh, come sempre dipende molto dalla situazione di utilizzo in cui ti trovi. Secondo me certamente in moltissimi casi è un'ottima soluzione avere contenitori leggeri invece di dover utilizzare un'intera macchina virtuale tutto il tempo, soprattutto se vuoi far girare le applicazioni efficientemente e usare meno risorse possibile, fattore che diventa particolarmente importante quando lavori nel cloud, naturalmente. Il fatto che sia un'ottima soluzione non significa però che non ci siano assolutamente applicazioni per l'altro metodo. Dipende da cosa devi fare.

LXP: Per la virtualizzazione su larga scala nel cloud, usando meno risorse possibile e sulla base del kernel, secondo te è effettivamente meglio?
KM: Sì, penso che sia un metodo di lavoro molto efficiente. LXP





Premiata Amministeria Dott. Brown

Jolyon Brown
Quando non fa consulenze su Linux, Jolyon passa il suo tempo lavorando alla sua grande ambizione: trovare un motivo per usare Emacs...

Tecniche esoteriche per i sysadmin dai recessi più impenetrabili della sala server

Il lupo perde il vizio?

Al recente ChefConf, che si è tenuto negli Stati Uniti, un rispettabile tecnico di Microsoft, Mark Russinovich, durante una conferenza sull'Open Source ha chiesto ai presenti quanti di loro usassero esclusivamente Windows per lo sviluppo: solo un coraggioso ha alzato la mano. È anche stato chiesto se mai esisterà un Windows Open Source e, secondo Russinovich: "Tutte le discussioni che potete immaginare sono già state fatte". Credo che dovrà passare ancora molto tempo prima che possiamo trovare Windows nel DVD allegato a LXP, ma bisogna dire che la nuova Microsoft guidata da Satya Nadella sembra adottare un nuovo approccio all'Open Source in generale. Per esempio, la sua nuvola **Azure** mette a disposizione un'opzione Linux; è possibile scaricare **.NET** da Github e l'azienda ha annunciato versioni per più piattaforme di **Visual Code Studio**. Come sappiamo non è andata sempre così: chiunque conosca la Microsoft dell'inizio degli anni '90 e dei primi anni 2000 si ricorderà di episodi come la pubblicazione dei documenti di Halloween (memorandum interni che spiegavano come Microsoft progettava di attaccare Linux) e di Steve Ballmer che usava parole come "comunismo" e "cancro" riferendosi a Linux e alla licenza GNU. All'epoca Microsoft era un gorilla aggressivo di 200 kg (qualcuno direbbe un gorilla anti-competitivo, dato che l'azienda corse il rischio di essere divisa in più società per ordine del giudice) che aveva distrutto tutti i concorrenti sul desktop (compreso il povero Netscape) ed era famosa per la sua filosofia *embrace, extend, extinguish*. Le cose sono ora completamente differenti e l'Open Source ha avuto un ruolo significativo nel modificare il paesaggio in cui si trova a operare Microsoft. Qualsiasi admin Linux che abbia da integrare un singolo sistema Windows accoglierà con favore questi sviluppi e si augurerà che l'azienda proceda un questa direzione. Se solo Windows avesse un server SSH decente...

Arriva una nuova release per OpenStack

OpenStack arriva alla versione **11** e **Ubuntu** annuncia un nuovo strumento per i contenitori

Il progetto OpenStack è riuscito nella difficile impresa di rispettare la data prevista per il rilascio di **Kilo**. Si tratta dell'undicesima iterazione della piattaforma, dedicata alla memoria di Chris Yeoh, un famoso contributore di OpenStack purtroppo scomparso in aprile. Sono state rilasciate circa 400 nuove funzionalità e corrette alcune migliaia di bug nell'intera piattaforma (che sta diventando sempre più grande). Questa release si concentra sulla stabilità e vede il completamento del componente che ha a che fare col "nudo metallo", cioè direttamente con l'hardware, (appropriatamente chiamato **Ironic**) che risulta ora essere pronto per la produzione ed è effettivamente utilizzato da alcuni fornitori di servizi cloud. Ironic è anche in grado di lavorare con macchine virtuali e ha adottato alcune tecnologie emergenti come i contenitori Linux, il *platform-as-a-service*e l'NFV (*Network Function Virtualization*: bilanciatori di carico virtualizzati, ecc.). Ubuntu ha inserito Kilo nella sua release 15.04 (**Vivid Vervet**). È compreso anche **LXD**, un nuovo ipervisore di contenitori che, secondo le dichiarazioni di OpenStack, è in grado di fornire prestazioni analoghe ai sistemi nativi, mettendo però a disposizione tutti gli strumenti di gestione e per la sicurezza indispensabili quando si ha a che fare con un certo numero di VM e di sistemi virtualizzati. È anche disponibile un driver per OpenStack il cui uso in produzione viene però scoraggiato. Ubuntu promette comunque di riuscire a farlo inserire nella distribuzione ufficiale di OpenStack. La 15.04 vede anche la prima release di **Snappy**, la versione di **Ubuntu Core** per la nuvola (anche se presumibilmente questa tecnologia è pensata per l'installazione su dispositivi mobili come gli smartphone). Si tratta della più piccola versione del sistema operativo di Canonical e fa uso degli aggiornamenti transazionali, una nuova tendenza dei sistemi operativi, che sono stati un beta per un certo periodo.
È una tecnologia molto diversa rispetto alla maniera tradizionale di distribuzione di patch e aggiornamento per un sistema operativo, che elimina completamente gli strumenti tradizionali come **apt-get**. Il nuovo sistema promette di realizzare installazioni e aggiornamenti con un singolo comando. Risulta essere fortemente influenzato dal modello dei contenitori, dato che il sistema operativo e le applicazioni vengono distribuiti come immagini, mentre gli aggiornamenti sono dei *delta* che vengono applicati alle immagini. Lo si può provare in un'ampia gamma di piattaforme, dalla Raspberry Pi a vari servizi cloud.

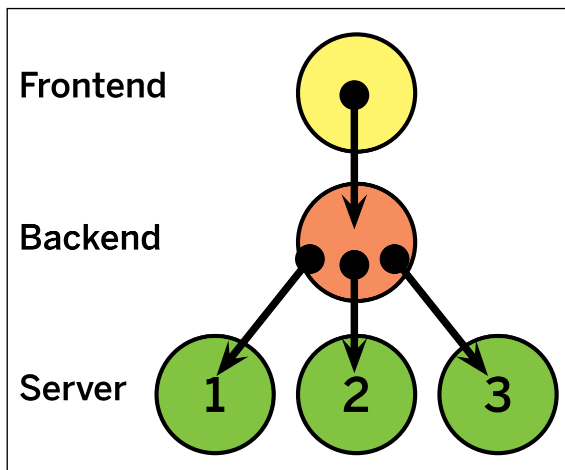


› **Il logo di *Kilo*, la più recente versione della suite *OpenStack***

CoreOS: il futuro è qui!

In questa seconda parte della nostra investigazione sulle meraviglie di **CoreOS** ci occuperemo di scoperta dei servizi e di clustering

Il mese scorso ho iniziato un esame di CoreOS, la distribuzione (distro) Linux che usa i contenitori per gestire i servizi e utilizza un sistema di aggiornamento automatico. Abbiamo visto come installare e configurare un sistema di test basato su **Vagrant**, abbiamo esaminato un po' il sistema di aggiornamento automatico e abbiamo visto alcuni dei comandi **fleet** (lo strumento di CoreOS per la gestione dei cluster). In questa seconda puntata continuerò cercando di capire come potrei usare CoreOS per creare una infrastruttura per un cliente. Questo mi aiuta a farmi un'idea di come i vari componenti lavorano insieme. Se non avete letto la prima parte della serie (LXP 158) vale la pena di dargli un'occhiata, soprattutto se volete riprodurre gli esempi che illustrerò più avanti. Nel box qui sotto (**Installare CoreOS su Vagrant**) potete trovare un riassunto dei comandi necessari per partire. Alla fine del mio ultimo esperimento avevo tre istanze di CoreOS pronte a gestire dei servizi su un cluster. Sto cercando di creare un tipico scenario che potrebbe essermi richiesto da un cliente: far girare un sistema di *hosting* resiliente su questo cluster. Proseguo quindi aggiungendo una ulteriore istanza di CoreOS per mostrare come **etcd** tratta l'improvvisa mancanza di disponibilità di uno o più nodi. Abbiamo bisogno di un quorum di almeno tre nodi per farlo, per questo è essenziale averne un quarto. Parto dall'ipotesi che il cluster sia giù (non credo che tutti quelli che hanno letto l'articolo del mese scorso abbiano lasciato girare le loro VM per un mese). In questo caso la prima cosa fare è ri-registrarsi con il servizio pubblico di scoperta di CoreOS (**https://discovery.etcd.io/new**). Anche questa volta dovete inserire il valore fornito dal servizio all'interno del file **user-data** che si trova nella stessa directory in cui si trova il vostro **Vagrantfile**, aggiornando l'URL sulla riga che inizia con **discovery**. Se avete installato CoreOS il mese scorso probabilmente verrete avvertiti da Vagrant stesso della disponibilità di una nuova versione. In questo caso basta un veloce `$ vagrant box update` per scaricare l'immagine più recente (nel mio caso la versione 647.0.0). Fatto questo modificate il valore di **num_instances** nel file **config.rb** per portarlo da tre a quattro. A questo punto il cluster può essere avviato con il comando `$ vagrant up`. Quasi immediatamente verranno avviati tutti e quattro i nodi



› ***Vulcand* instrada le connessioni verso il server di backend corretto confrontando il percorso nell'URL della richiesta con le informazioni fornite da *etcd***

e risulteranno online. Ora mi collego con SSH al primo (`$ vagrant ssh core-01`) e verifico velocemente lo stato del cluster con `$ fleet list-machines`

MACHINE	IP	METADATA
462cf0c6...	172.17.8.102	-
8841d0cb...	172.17.8.101	-
af2fde58...	172.17.8.104	-
cd8df6e3...	172.17.8.103	-

Vulcand: come bilanciare il carico

Ho intenzione di utilizzare **fleet** per installare un'istanza di **vulcand** (**http://www.vulcanproxy.com/**), un proxy per microservizi e gestione di API, che io utilizzerò però in maniera più tradizionale. La cosa interessante di questo software, scritto da un'azienda di nome Mailgun (e Open Source), è che usa **etcd** come *backend* di configurazione piuttosto che i più tradizionali file di configurazione utilizzati per esempio da **HAProxy**. È possibile usare quest'ultimo per fare qualcosa di simile usando però delle estensioni di terze parti e con un po' di lavoro in più. La mia intenzione è di fare in modo che **vulcand** registri automaticamente le istanze delle applicazioni di cui **fleet** fa il deploy e ridiriga il traffico verso di esse senza necessità di un riavvio o della modifica dei file di configurazione. Le applicazioni gireranno tutte all'interno di

»

Installare CoreOS su Vagrant

Se vi siete persi l'ultimo numero non sarà facile seguire gli esempi di questo mese. Ecco un riassunto veloce che vi permetterà di mettervi in pari. Per prima cosa scaricate la versione più recente di Vagrant da **http://www.vagrantup.com** (che è leggermente più aggiornato della versione che sto usando io su Ubuntu 14.04). Io ho installato anche **VirtualBox**, che rende un po' più semplice l'utilizzo di Vagrant, e **Git** per lavorare sul relativo repository. Fatto questo date i seguenti comandi:

```
$ git clone https://github.com/coreos/coreos-vagrant.git
$ cd coreos-vagrant
$ mv user-data.sample user-data
$ mv config.rb.sample config.rb
```

Ora procuratevi un token visitando **http://discovery.etcd.io**. Rimuovete il segno di commento all'inizio della riga **discovery** in **user-data** e sostituite la stringa **https://discover.etcd.io** presente sulla riga con il contenuto della finestra del browser. Aprite poi **config.rb**, eliminate il segno di commento all'inizio e aggiornate le righe contenenti **num_instances** e **update_channel** come mostrato sotto:

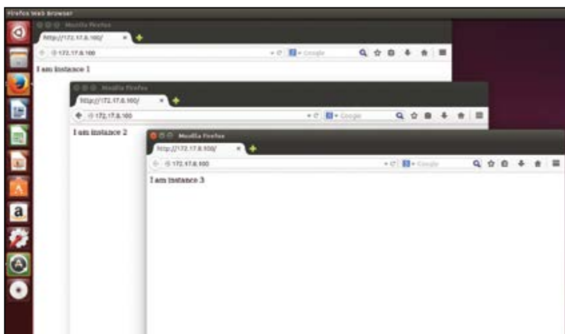
```
$num_instances=3
$update_channel='stable'
```

Infine attivate vagrant e il cluster di test CoreOS. Verrà scaricato il "box" CoreOS di Vagrant e poi tutto partirà: `$ vagrant up`.

» contenitori Docker e **fleet** si occuperà di garantire la loro alta disponibilità, rieseguendo il deploy se io eseguo il riavvio del rispettivo host. È possibile, e probabilmente preferibile, controllare **fleet** da un client locale (cioè dalla macchina che sto usando come desktop), creando un tunnel SSH verso i sistemi CoreOS. In un ambiente di produzione faccio proprio così (usando per il "salto" un host che possibilmente si trovi all'interno di un segmento di rete sicuro che richiede un'autenticazione a due fattori per l'accesso). Ma qui si tratta semplicemente di prove, con poco tempo e scarsa disponibilità di mezzi, perciò controllerò **fleet** direttamente dal primo dei nodi del cluster. La prima cosa da fare è installare e attivare **vulcand** sul mio cluster, quindi creerò una sottodirectory che chiamerò **units** e al suo interno il file **vulcand.service** con il seguente contenuto:

```
[Unit]
Description=Vulcan
After=docker.service

[Service]
Restart=always
TimeoutStartSec=0
ExecStartPre=-/usr/bin/docker kill vulcan1
ExecStartPre=-/usr/bin/docker rm vulcan1
ExecStartPre=/usr/bin/docker pull mailgun/vulcand:v0.8.0-beta.2
ExecStartPre=/usr/bin/sudo /usr/bin/ip addr add 172.17.8.100/24 dev eth1
ExecStartPre=/usr/bin/etcdctl set /vulcand/backends/lxp/backend '{"Type": "http"}'
ExecStartPre=/usr/bin/etcdctl set /vulcand/frontends/f1/frontend'{"Type": "http", "BackendId": "lxp", "Route": "Path(`/`)"}'
ExecStart=/usr/bin/docker run --rm --name vulcan1 -p 80:80 -p 8182:8182 mailgun/vulcand:v0.8.0-beta.2 /go/bin/vulcand -apiInterface=0.0.0.0 -interface=0.0.0.0 -etcd=http://10.1.42.1:4001 -port=80 -apiPort=8182
ExecStopPost=-/usr/bin/sudo /usr/bin/ip addr del 172.17.8.100/24 dev eth1
ExecStop=-/usr/bin/docker stop vulcan1
```



› **Ora che sono riuscito a instradare in maniera resiliente il mio traffico su più server di backend devo solo trovare un'applicazione che faccia davvero qualcosa...**

Come abbiamo già visto nel primo articolo, si tratta di un file di unità di Systemd che definisce come verrà eseguito il mio servizio **vulcand**. Esaminandolo più da vicino vediamo che le prime due righe semplicemente danno il nome all'unità e dicono a Systemd che può girare solo dopo che che il servizio Docker è partito. Restart dice a Systemd che voglio che questo servizio ritorni in linea nel caso di un'interruzione inaspettata (da notare che un arresto ordinato del sistema non causa la ripartenza automatica). Le righe ExecStartPre fanno un po' di pulizia delle vecchie istanze e assegnano un indirizzo IP all'interfaccia eth1 della macchina sulla quale **fleet** deciderà di far girare il servizio (ExecStopPost cancella l'indirizzo). Questo significa che potrò sempre contattare **vulcand** al medesimo indirizzo IP. Il - all'inizio delle righe docker kill e rm fa in modo che non vengano generati messaggi di errore se l'istanza Docker in questione non esiste. In caso contrario Systemd non sarebbe in grado di attivare l'unità. La riga che contiene edcdctl aggiunge la definizione di un frontend e di un backend a **etcd** (nota: è richiesto il percorso assoluto dei binari, in caso contrario la riga viene ignorata). La riga ExecStart che segue è quella che effettivamente fa partire **vulcand** all'interno di un contenitore Docker e apre le porte. Viene anche passato l'indirizzo IP e la porta per connettersi a **etcd**. I dati sono mostrati dal comando

```
$ ip addr show
```

Per un contenitore si tratta dell'indirizzo dell'interfaccia docker0. Dato che stiamo usando Vagrant potrebbe trattarsi di un indirizzo diverso da quello usato in una installazione standard di CoreOS. Vi conviene controllare quale sia quello effettivamente usato sul vostro sistema e correggere di conseguenza il file di configurazione dell'unità. Senza ulteriori indugi posso ora lanciare **vulcand** sul mio cluster e verificarne lo stato. **Fleet** deciderà su quale nodo farlo girare, noi non dobbiamo preoccuparci di questo. Se si trattasse di un sistema di produzione avrei probabilmente più gruppi di sistemi CoreOS disposti su segmenti diversi della rete per ragioni di sicurezza, ma ora sto semplicemente cercando di capire come funziona il prodotto.

```
$ fleetctl start vulcand.service
Unit vulcand.service launched on 8841d0cb.../172.17.8.101
$ fleetctl list-units
UNIT            MACHINE                  ACTIVE    SUB
vulcand.service    8841d0cb.../172.17.8.101 active    running
```

Per ora sembra tutto a posto. Se visito **http://172.178.8.100**,

Guerre tra contenitori e flanelle

Per puro caso mentre stavo scrivendo questo articolo ha avuto luogo il **CoreOS Fest**, nel corso del quale sono stati fatti alcuni annunci, in particolare riguardo le specifiche di contenitore di CoreOS, che sono in contrapposizione con quelle di Docker. La specifica di App Container **appc** (di cui **rkt**, di CoreOS, è un'implementazione) ha ora numerosi manutentori non affiliati a CoreOS: rappresentanti di Google, Twitter e Red Hat anno ora la possibilità di influire sulla direzione futura delle specifiche. Google ha implementato **rkt** all'interno di Kubernetes, mentre VMware l'ha inserito in Projet Photon, la sua implementazione leggera di Linux. Apcera, un'azienda che si rivolge a utenti enterprise di "nuvole ibride", ha scritto (e reso Open Source) la sua implementazione, nota come **Kurma**. Sembra che gli schieramenti siano decisi, anche se più o meno tutti cercano di ridurre i rischi supportando sia Docker sia tutti gli altri tipi di contenitore. CoreOS ha anche fornito qualche dettaglio in più sui suoi piani per il networking. È già possibile scaricare **flannel**, un'implementazione del modello Kubernetes in cui a ciascuna macchina di un cluster viene assegnata un'intera sottorete. Questo permette di ridurre la complessità nel mappaggio delle porte, ma per ora tra i fornitori di servizi cloud solo Google è in grado di farlo. **flannel** usa il dispositivo universale TUN/TAP e crea su di esso una rete usando UDP per incapsulare i pacchetti IP. Si può scaricare da Github, se l'argomento vi stimola, in **https://github.com/coreos/flannel** troverete i sorgenti e anche alcuni utili diagrammi.

l'indirizzo IP assegnato prima, vedrò il messaggio di errore **{"error": "not found"}**, dato che per ora non c'è nessun carico da bilanciare.

Frontend e backend

Vulcand, come molti altri bilanciatori di carico, per funzionare ha bisogno della definizione di servizi di *frontend* e servizi di *backend*. La cosa bella nel nostro caso è che, anziché modificare i file di configurazione, le informazioni saranno scritte in **etcd**. E, ancora meglio, avverrà tutto in automatico. Questo richiede un po' più di lavoro all'inizio, ma ne vale la pena. Nel file di unità di **vulcand** abbiamo già definito un backend chiamato **lxp**, di tipo HTTP, e un frontend chiamato **f1** che invia tutto il traffico (indicato dal percorso di contesto **/**) al backend lxp. Se dovessimo far girare più applicazioni potremmo definire backend differenti, corrispondenti a diversi percorsi di contesto (per esempio **/pagamenti/** oppure **/login/**). Restando nella directory **units** creerò ora altri tre file chiamati **hosta.service**, **hostb.service** e **hostc.service** (potete trovarli su **http://pastebin.com/jPiM5adb**).

Ecco il primo:

```
[Unit]
Description=Nginx
After=docker.service

[Service]
EnvironmentFile=/etc/environment
TimeoutStartSec=0
ExecStartPre=-/usr/bin/docker kill nginx1
ExecStartPre=-/usr/bin/docker rm nginx1
ExecStartPre=/usr/bin/docker pull nginx
ExecStartPre=/bin/sh -c "mkdir -p /home/core/www; echo 'I
am instance 1' > /home/core/www/index.html"
ExecStartPre=/usr/bin/etcdctl set /vulcand/backends/lxp/
servers/srv1 '{"URL": "http://${COREOS_PUBLIC_
IPV4}:8080"}'
ExecStart=/usr/bin/docker run --name nginx1 -p 8080:80 -v /
home/core/www:/usr/share/nginx/html nginx
ExecStopPost=/usr/bin/etcdctl rm /vulcand/backends/lxp/
servers/srv1
ExecStop=-/usr/bin/docker stop nginx1

[X-Fleet]
Conflicts=host*.service
```

Ed ecco **hostb.service**:

```
[Unit]
Description=Nginx
After=docker.service

[Service]
EnvironmentFile=/etc/environment
TimeoutStartSec=0
ExecStartPre=-/usr/bin/docker kill nginx2
ExecStartPre=-/usr/bin/docker rm nginx2
ExecStartPre=/usr/bin/docker pull nginx
ExecStartPre=/bin/sh -c "mkdir -p /home/core/www; echo 'I
am instance 2' > /home/core/www/index.html"
ExecStartPre=/usr/bin/etcdctl set /vulcand/backends/lxp/
servers/srv2 '{"URL": "http://${COREOS_PUBLIC_
IPV4}:8080"}'
ExecStart=/usr/bin/docker run --name nginx2 -p 8080:80 -v
/home/core/www:/usr/share/nginx/html nginx
ExecStopPost=/usr/bin/etcdctl rm /vulcand/backends/lxp/
servers/srv2
ExecStop=-/usr/bin/docker stop nginx2

[X-Fleet]
Conflicts=host*.service
```

Concludiamo con **hostc.service**:

```
[Unit]
Description=Nginx
After=docker.service

[Service]
EnvironmentFile=/etc/environment
TimeoutStartSec=0
ExecStartPre=-/usr/bin/docker kill nginx3
ExecStartPre=-/usr/bin/docker rm nginx3
ExecStartPre=/usr/bin/docker pull nginx
ExecStartPre=/bin/sh -c "mkdir -p /home/core/www; echo 'I
am instance 3' > /home/core/www/index.html"
ExecStartPre=/usr/bin/etcdctl set /vulcand/backends/lxp/
servers/srv3 '{"URL": "http://${COREOS_PUBLIC_
IPV4}:8080"}'
ExecStart=/usr/bin/docker run --name nginx3 -p 8080:80 -v /
home/core/www:/usr/share/nginx/html nginx
ExecStopPost=/usr/bin/etcdctl rm /vulcand/backends/lxp/
servers/srv3
ExecStop=-/usr/bin/docker stop nginx3

[X-Fleet]
Conflicts=host*.service
```

Ho aggiunto la riga EnvironmentFile per Systemd, cosa che mi permette di utilizzare COREOS_PUBLIC_IPV4 più avanti nel file, all'interno del comando edcdctl, per aggiungere ciascun server al backend lxp. L'ultima riga riguarda **fleet** e lo informa che ciascuna versione di queste unità non può essere eseguita su un nodo su cui sia già in esecuzione una delle altre. Quando le faccio partire vengono perciò distribuite su tutto il cluster:

```
$ fleetctl start host*.service
$ fleetctl list-units
UNIT           MACHINE                 ACTIVE    SUB
hosta.service  af2fde58.../172.17.8.104  active  running
hostb.service  cd8df6e3.../172.17.8.103  active  running
hostc.service  8841d0cb.../172.17.8.101  active  running
vulcand.service 8841d0cb.../172.17.8.101 active  running
```

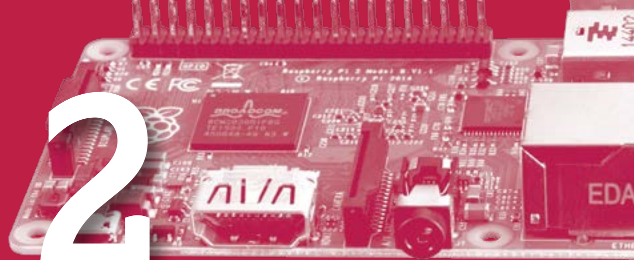
Se ora mi collego con **http://172.178.8.100** vedrò una pagina che mi mostra a quale istanza mi ha diretto **vulcand**. Se forzo un nuovo caricamento della pagina (evitando la cache del browser) vedrò un numero diverso. Posso anche verificare il contenuto della cache di **etcd** usando **curl**:

```
$ curl -L http://127.0.0.1:4001/v2/keys/vulcand/backends/lxp/
servers
```

Verranno visualizzati un certo numero di dati JSON che mostrano i server lxp registrati. Se ora arresto uno di questi backend:

```
$ fleetctl stop hostc.service
```

e ripeto il comando **curl**, i dati corrispondenti al server fermato saranno spariti: **vulcand** non indirizzerà più il traffico verso quel server. Il comando list-units di fleetctl mostrerà il server fermato come **inactive/dead**: è normale. Analogamente, forzando l'arresto di **vulcand** (con **kill -9** o spegnendo il server su cui sta girando) si obbligherà **fleet** a farlo ripartire su un altro nodo. LXP



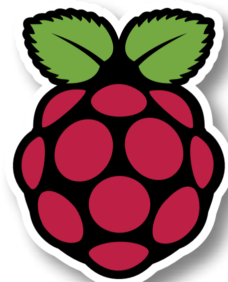


LIZ UPTON ci dà le ultime notizie dalla Raspberry Pi Foundation.

Benvenuti...

Alla Pi Towers organizziamo stage per studenti. Sia la Fondazione sia la parte commerciale dell'organizzazione, Raspberry Pi (Trading) Ltd, sono sempre alla ricerca di insiemi di capacità diverse negli studenti che scelgono. Raspberry Pi Trading, per esempio, cerca persone con conoscenze avanzate di linguaggio assembly a basso livello, perché interverranno sul kernel e impareranno a lavorare su circuiti integrati ASIC. Uno dei nuovi stagisti aveva già scritto il suo compilatore personale a casa nei momenti liberi e siamo entusiasti di vedere cosa realizzerà nel corso della sua permanenza con noi. Abbiamo anche preso 2 studenti che avevano realizzato progetti di grande spicco. Tom Hartley, noto per AirPi, la stazione meteorologica di Raspberry Pi, realizzerà demo che potremo usare per affascinare le persone agli eventi, e Andrew Mulholland farà dei ritocchini al sistema PiNet che consente agli insegnanti di gestire facilmente reti Pi in classe. Paghiamo ai nostri stagisti lo stesso stipendio dei nostri programmatori neoassunti... e nessuno gli fa preparare il caffè o cose del genere. Se siete interessati a uno stage alle Pi Towers per l'estate del 2016, mandateci un curriculum agli indirizzi che trovate sul sito. Negli ultimi tempi abbiamo anche lanciato il Raspberry Pi case e siamo rimasti soddisfatti delle vendite: ottenere un risultato ottimale sin nei minimi dettagli ci è costato molto e siamo felici che sia stato apprezzato. Se volete saperne di più visitate **https://www.raspberrypi.org/products/raspberry-pi-case/**.

I vincitori del progetto Astro Pi



La Pi Foundation sceglie i migliori progetti

La notizia che la Pi Foundation avrebbe mandato **Astro Pi** sulla Stazione Spaziale Internazionale (ISS) con l'astronauta britannico Tim Peake è stata fantastica, ma ancora di più ci ha entusiasmato sapere che Astro Pi avrebbe lavorato con codice scritto dai ragazzi delle scuole per fare esperimenti nello spazio. Dopo aver ricevuto tutte le candidature dei vari progetti, un gruppo di esperti che comprendeva l'Agenzia Spaziale britannica e ESERO UK si è incontrata alle Pi Towers e ha esaminato tutto il codice su un vero Astro Pi. Gli esperimenti che hanno vinto includevano test creativi come usare la videocamera della Pi per individuare le radiazioni cosmiche realizzato dal Magdalen College School, o il progetto di Thirsk School che usa dati offline per geolocalizzare l'ISS, predire su che Paese si trova e visualizzarne la bandiera. C'è persino un progetto basato su ***Minecraft*** che usa il gioco per visualizzare i set di dati del sensore. Tutto il codice che abbiamo ricevuto andrà sulla ISS che può ricevere 10 schede SD, ed è possibile che Peake usi anche qualcuno dei programmi che non hanno vinto. Più dettagli su: **http://astro-pi.org**.



SSD PiDrive

Tanta capacità, poco ingombro

Aggiungere un hard disk esterno a Raspberry Pi è piuttosto facile ma ha i suoi inconvenienti perché lo rende più ingombrante. Ora però è in fase di sviluppo un nuovo dispositivo, il **PiDrive**, che ci permette di eliminare drive esterni e cavi di collegamento. Chi ha una Raspberry Pi può infatti sfruttare questa soluzione di salvataggio sottile ma di grande capacità. Il PiDrive si monta direttamente sul connettore di Input/Output a 40-pin che si trova su tutte le Raspberry Pi B+, A+, o B+ v2, poi si può installare qualsiasi unità a stato solido SATA sulla Raspberry Pi, ottenendo velocità operative molto più elevate e una capacità maggiore che con le opzioni di salvataggio tradizionali per le Raspberry Pi. Il progetto ha avuto un grande successo su Kickstarter, raggiungendo l'obiettivo in soli 3 giorni, e ora siamo in attesa del lancio!



› Aggiungete un card di espansione SSD ad alta capacità alla vostra Raspberry Pi

Rapid Radio

Comunicazioni radio in arrivo!

Raspberry Pi può usare Bluetooth e Wi-Fi per comunicare e ora un add-on dal nome di **Rapid Radio** permette anche di aggiungere comunicazioni radio ai progetti per Pi (o Arduino). La radio ha un ricetrasmettitore half duplex a 2,4 GHz, una velocità massima di 2 Mbit/s e un raggio di circa 70 metri all'aperto e un po' di più in interni. Ha anche un bellissimo aspetto, dato che il circuito stampato ha la forma di una farfalla. Il suo punto di forza, però, è la facilità con cui consente di aggiungere la comunicazione radio a Raspberry Pi. L'intero codice sorgente per integrarla in qualsiasi progetto è disponibile su GitHub, e si può usare e modificare liberamente con la licenza MIT. Raccolti i fondi necessari tramite Kickstarter, Rapid Radio è ora in produzione e potete prenotarlo su **http://rapidradio.pl/buy/**.



› Funzioni di comunicazione radio facili per Raspberry Pi

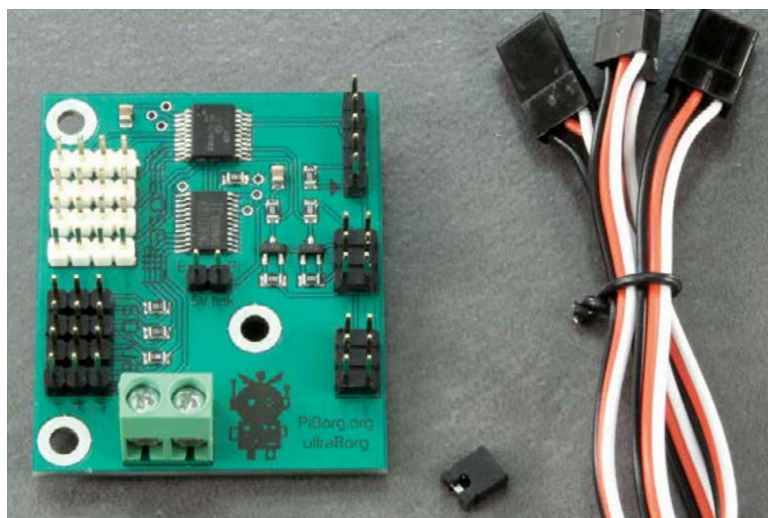
Ultraborg

Cercando di non seguire la tradizione dei Terminator, il nostro robot si evolverà da una RP e non da Skynet grazie a questa scheda piena di controlli per sensori

In breve

» *Ultraborg* è una scheda aggiuntiva per PiBorg e può essere usata per il controllo di sensori ultrasonici multipli e per servomotori impiegati nella costruzione di robot. In sostanza è un microcontrollore di semplice utilizzo in grado di sgravare la CPU della RP dai compiti di gestione degli input e di controllo.

PiBorg è un'azienda che da tempo realizza schede d'espansione per la RP e Arduino; la loro creazione più recente si chiama **Ultraborg**. Si tratta di un add-on con interfaccia I2C per tutti i modelli di Raspberry Pi, Arduino e altre schede compatibili con la stessa interfaccia di comunicazione. Il solido PCB è in grado di controllare quattro unità servo – motori controllabili con estrema precisione che vengono utilizzati per muovere i robot o le telecamere, per esempio, e accetta fino a quattro sensori ultrasonici HC-SR04 in input. Questi sensori economici sfruttano un fascio di onde sonore per identificare la presenza di ostacoli lungo il percorso. Grazie al PIC presente sulla scheda, è possibile leggere gli input di tutti e quattro i sensori contemporaneamente sgravando dal compito il SoC della RP (o il microcontrollore di Arduino). Esistono due modi per alimentare la scheda Ultraborg: se dovete usare solo i sensori ultrasonici sono sufficienti i 5v provenienti dal GPIO della RP; se invece dovete impiegare anche dei servo, è necessario usare un'alimentazione esterna, sempre da 5v, collegata a dei morsetti a vite. Volendo potete anche acquistare la scheda BattBorg (che consente di alimentare la RP usando delle normali pile AA). L'installazione del software necessario all'Ultraborg si esegue in due passaggi sulla RP: prima bisogna configurare il bus I2C sul piccolo computer usando **raspi-config**. Il secondo passo consiste invece nel download dello script d'installazione dal sito di PiBorg, che installerà anche diversi script d'esempio. Il primo di essi dimostra come si usano dei servo motori in sequenza, cosa utile per verificare il corretto funzionamento di un vostro progetto che usa questo tipo di motori. Ovviamente ci sono anche degli script d'esempio che illustrano il funzionamento dei sensori ultrasonici.



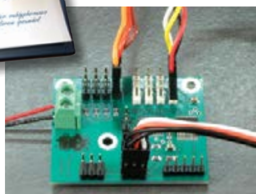
› *Ultraborg* **racchiude in pochissimo spazio tantissimi componenti. Il PCB appare ben progettato e costruito in modo solido**

Ultrasuoni semplificati

I file d'esempio includono anche alcune applicazioni dotate d'interfaccia grafica da usare per verificare e impostare in modo fine il funzionamento della scheda all'interno dei vostri progetti. Fornendo all'utente una libreria Python di facile impiego, il team di PiBorg ha semplificato enormemente il compito di controllare i sensori ultrasonici, che di solito richiedono alcuni calcoli matematici per determinare la distanza degli ostacoli. Allo stesso modo è reso semplice anche l'uso dei servo, che richiedono la conoscenza della PWM (*Pulse Width Modulation*) per il loro controllo. Inoltre il software non si limita alla Raspberry Pi, ma contiene anche le librerie equivalenti per l'uso di Ultraborg in un progetto basato su Arduino.

Il prezzo base è di circa 22 euro fino ad arrivare a circa 118 euro per il kit completo, che include quattro sensori HC-SR04 e quattro grandi servo motori, assieme a tutti i cavi necessari ad alimentare e controllare i componenti. Forse la cifra per il kit completo può sembrare alta, ma è giustificata perché grazie a questa scheda potrete realizzare progetti molto sofisticati. LXP

Caratteristiche



Microcontrollore

Il microcontrollore PIC si occupa di gestire gran parte dei calcoli più critici al posto della scheda host a cui è collegato.



Input e output

È in grado di far funzionare quattro sensori ultrasonici HC-SR04 alla volta, un ottimo risultato per una scheda così piccola.

Linux Pro Giudizio

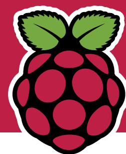
Ultraborg

Sviluppatore: Piborg
Web: www.piborg.org/ultraborg
Prezzo: 22,00 € + spese di spedizione

Caratteristiche	8/10
Prestazioni	8/10
Facilità d'uso	8/10
Qualità/Prezzo	8/10

» *La potenza del microcontrollore incluso consente esperimenti di robotica anche con le RP più vecchie.*

Il voto di Linux Pro 8



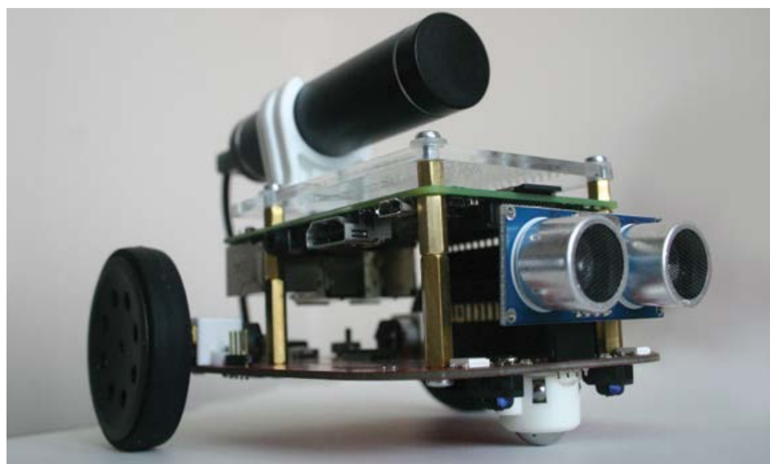
Agobo 2

Emulando il lavoro del dottor Frankenstein la redazione di LXP si è divertita a montare questo robot, più ricco di metallo che di bit

In breve

» Una semplice ma anche espandibile piattaforma per la costruzione di robot, progettata per funzionare con i modelli A+, B+ e 2 della Raspberry Pi. Pensata per i principianti, include un PCB ben studiato e tutte le componenti per l'assemblaggio di un robot. A corredo c'è una buona libreria Python che consente il controllo di quanto realizzato in modo sufficientemente semplice.

I progetti di robotica amatoriale possono portare a risultati singolari ma non sono certo banali da realizzare. Per fortuna esistono aziende come 4tronix che mettono a disposizione di tutti dei kit in grado di farci arrivare alla meta desiderata. L'Agobo originale fu rilasciato alla fine del 2014 ed era composto da un PCB, che costituiva l'intelaiatura del robot, e da componenti presaldati ai quali l'utente poteva collegare una Raspberry Pi A+. Proprio l'obbligo di usare una scheda A+ costituì una delle maggiori critiche mosse dalla comunità, perché questo modello di RP non era certo il più diffuso. Ascoltando questi commenti, 4tronix ha quindi deciso di rinnovare il suo robot creandone una seconda versione compatibile con i modelli A+, B+ e 2. La nuova disposizione delle connessioni e dei componenti consente di montare la RP mantenendo le porte USB sul retro del robot e l'uso di distanziatori aggiuntivi fa sì che si possano usare le RP più nuove.
Il produttore è ben noto per usare il PCB come "corpo" del robot e così facendo fornisce una solida piattaforma mettendo tutta l'elettronica necessaria in un unico pacchetto. Agobo 2 usa un micromotoriduttore in metallo, assieme al controller per motori integrato, permettendo un controllo bidirezionale del movimento del robot. Gli input sono gestiti da un sensore ultrasonico HC-SR04 e da due sensori line-following, in grado di identificare una linea disegnata davanti al robot, facendogliela quindi seguire. Visto che di solito un robot non dispone di uno schermo, in genere ci si collega a esso via SSH, usando il Wi-Fi o Ethernet; Agobo 2, invece, ha un'interfaccia seriale dedicata da usare con un cavo USB TTL, permettendo quindi il collegamento al robot direttamente dalla porta USB di un computer. Agobo 2 ha anche un piccolo interruttore che consente di dar vita al progetto, grazie a una libreria Python. Ad alimentare il tutto c'è una piccola batteria da cellulare, collegata al PCB e in grado di fornire la giusta alimentazione al robot e alla RP.



› **Il PCB di *Agobo 2* è stato sapientemente progettato e riesce a raggruppare tutti i componenti necessari al robot in un piccolo spazio**

Robot intelligente

Non basta l'hardware, ovviamente. Un buon robot necessita del giusto software per funzionare a dovere, e 4tronix ha messo assieme un'ottima libreria Python che controlla tutte le funzionalità del suo prodotto. La funzione **getDistance()** gestisce i complessi calcoli necessari per valutare la distanza di un oggetto usando gli ultrasuoni generati dai sensori ultrasonici. Esiste anche una funzione per l'uso dei sensori *line-following* che restituisce il loro stato attuale. Stesso discorso per il controllo dei motori: apposite funzioni consentono gli spostamenti in avanti e all'indietro, così come la rotazione. L'uso di queste funzioni è facile anche per i programmatori più giovani o meno esperti visto che nelle nostre prove sono bastate 14 linee di codice per creare un robot in grado di evitare ogni ostacolo posto ad almeno 50 cm di distanza dai sensori ultrasonici. Agobo 2 raffina la piattaforma originaria mantenendone però l'estrema facilità d'uso, sia dal punto di vista hardware che da quello software. Il kit richiede solo un cacciavite per essere montato e la libreria Python semplifica enormemente il lavoro di programmazione. LXP



Caratteristiche



PCB e chassis
Lo chassis riesce a ospitare tutti i componenti del robot, compresa la Raspberry Pi, le batterie per l'alimentazione e i motori.



Connessioni seriali
Per controllare il robot i progettisti hanno pensato bene di includere una connessione seriale tramite un cavo USB TTL.

LINUX PRO Giudizio

Agobo 2

Sviluppatore: 4tronix
Web: 4tronix.co.uk/store
Prezzo: 45 € (più spese di spedizione)

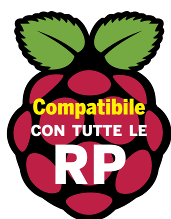
Caratteristiche	9/10
Prestazioni	10/10
Facilità d'uso	8/10
Qualità/Prezzo	9/10

» *Lo chassis è robusto e il tutto è facile da montare. La libreria fornita facilita il lavoro di controllo a tutti.*

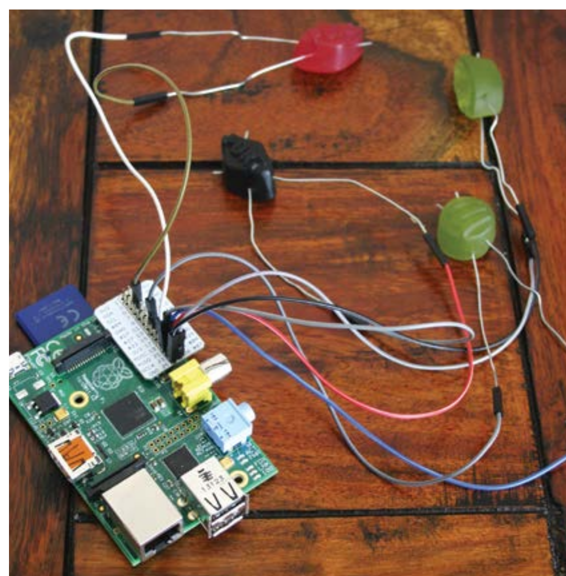
Il voto di Linux Pro 9

Scratch: costruite un controller

Linux Pro libera la sua fantasia usando una Raspberry Pi e qualche dolcetto per realizzare un controller per videogiochi



Per questo tutorial abbiamo pensato di creare un controller personalizzato per un gioco fatto con **Scratch**. Per realizzarlo vi serve il materiale seguente: un qualunque modello di Raspberry Pi; ScratchGPIO7 e Raspbian installati sulla RP; quattro caramelle gommose; otto graffette per carta; otto cavi elettrici femmina/femmina. Per la parte software, potete scaricare il codice dall'URL **http://bit.ly/LXP159_controller**. Vi starete chiedendo a cosa servono i dolcetti... no, non li dovete mangiare mentre programmate: userete le caramelle gommose come interfaccia di gioco. Abbiamo fatto diverse prove e quelle gommose si sono dimostrate quelle che forniscono la migliore interfaccia. Vanno bene anche le gelatine alla frutta, ma si deteriorano velocemente. Dovete inserire dentro ogni caramella due graffette opportunamente "smontate", cioè allungate, in modo che siano quasi parallele ma che, all'interno della caramella quasi si tocchino. Lasciate che le graffette sbuchino per circa 3 cm e a ogni estremità libera collegate un cavo femmina/femmina che, dall'altra parte, andrà collegato al GPIO della RP. I pin del GPIO da usare sono i seguenti: UP va nel pin 21; DOWN nel pin 22; LEFT nel pin 24 e, infine, RIGHT corrisponde al pin 26. Per ogni pulsante vi serve poi anche il collegamento a massa (GND) che potete trovare nei pin 6, 9, 14 e 20. Quando premete le caramelle, le due graffette si devono toccare. Visto che una di esse è collegata a un input che, di default è impostato al valore alto (1), mentre l'altra è a massa, quando le due graffette si toccano si innesca un cambiamento di stato e il valore 1 diventa 0: questo cambiamento può essere sfruttano in Scratch per eseguire il comando corrispondente. A questo proposito, prima abbiamo nominato ScratchGPIO7: si tratta di una versione modificata di Scratch in modo che possa interagire con il GPIO della Raspberry Pi. Non viene installata



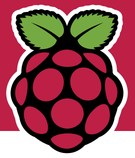
di default in Raspbian, ma potete farlo voi con facilità grazie a un particolare script. Con la RP connessa a Internet, aprite un terminale e digitate

```
$ wget http://bit.ly/1wxrqdp -O isgh7.sh
$ sudo bash isgh7.sh
```

Si installano due "versioni" di ScratchGPIO7: ScratchGPIO7 Plus che può essere usata in combinazione con varie schede aggiuntive per la RP, mentre ScratchGPIO7 serve quando si realizzano semplici circuiti con il GPIO. Avviate quest'ultima versione: l'interfaccia che appare è identica a quella di Scratch 1.4, con la palette di comandi sulla sinistra, divisa per gruppi di diversi colori in base alle funzioni fornite. I blocchi presenti sulla sinistra vanno trascinati nel pannello centrale e collegati

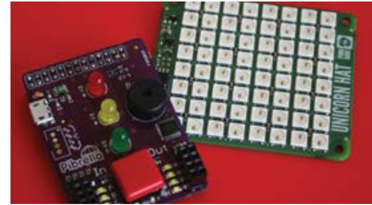
Schede aggiuntive

ScratchGPIO è un'interessante piattaforma per fare esperimenti con il GPIO della RP. Esso, difatti, fornisce anche ai programmatori più giovani un modo semplice per usare il pettine di pin del piccolo computer, e quindi muovere i primi passi nel mondo dell'elettronica, senza dover avere conoscenze particolari. Inoltre è un valido aiuto anche nel campo della robotica grazie al suo supporto per diverse schede aggiuntive. Board come Pibrella e RyanTeck Motor Control Board sono controllabili da ScratchGPIO, per esempio. Ma lo stesso vale per soluzioni più complesse come Pi2Go e Agobo di 4tronix. Per connettervi a una di queste schede dovete installare la libreria Python richiesta, aprire **ScratchGPIO** e creare una variabile chiamata **AddOn**, usando il blocco **porta AddOn a 0** e cambiando lo **0** con il valore relativo all'add-on che state usando, per esempio **Pibrella**. Ora vi basta seguire la guida che si trova all'URL **http://bit.ly/CymplecyAddOns** per iniziare i vostri esperimenti. Alcune board, però, non sono pienamente compatibili con ScratchGPIO sulla RP 2, mentre con le altre versioni non dovrebbero esserci problemi.



L'elettronica per tutti!

L'elettronica è un hobby che, di norma, si pensa sia alla portata degli adulti o comunque dei ragazzi abbastanza grandi. Invece, anche grazie a ScratchGPIO, anche i più piccoli possono vivere l'esperienza di creare il proprio hardware. Componenti come LED e resistori sono piccoli, ma con il supporto dei genitori e di poche istruzioni, i bambini possono imparare come usare questi componenti in modo sicuro. Potete poi acquistare (senza spendere molto) anche schede aggiuntive come Pibrella, UnicornHAT e TrafficHAT, che forniscono un'interfaccia hardware semplificata.



› Esistono molte schede aggiuntive per la RP, molte compatibili con ScratchGPIO

assieme in modo da formare una sequenza di istruzioni che comporranno il vostro progetto. Nella parte destra della finestra, invece, c'è un'ampia zona, chiamata stage, in cui appare l'output del codice mentre subito sotto di essa è presente l'elenco degli sprite, cioè gli oggetti usati nello stage. Per verificare il funzionamento del controller, provate a creare un programma che controlli il movimento del gatto, uno degli sprite più usati in Scratch. Il gioco dovrà permettervi di spostare l'animale in modo da evitare lo scontro con un altro sprite. Il codice inizia abilitando i pulsanti e usandoli per dare l'ordine al gatto di muoversi. Assicuratevi che lo sprite del gatto sia evidenziato e poi costruite il codice nella zona centrale; il codice per il pulsante **UP** è il seguente:

```
Quando si clicca su bandiera verde
 per sempre
   se valore del sensore pin21 = 0
      cambia y di 10
      attendi 0.1 secondi
```

Per l'istruzione **se** (**if** in inglese) bisogna usare il blocco **=** presente nella palette **Operatori** e al suo interno dovete posizionare il blocco **valore del sensore**, che si trova invece nella palette **Sensori**. Per assegnare il numero del pin corretto a quest'ultimo blocco è necessario cliccare sul menu a discesa e selezionare il numero corretto in base a quanto definito prima. Una volta che il pin in input è attivato, cosa che avviene trascinando il blocco nella giusta zona del codice, esso sarà subito impostato al valore 1, alto. Quando poi le due graffette si toccheranno, quella collegata a massa cambierà a 0 il valore di questo pin, attivando il codice corrispondente che farà spostare il gatto (lo sprite) modificando il suo valore dell'asse y. Assicuratevi che il pulsante **UP** sia collegato al pin 21 e a una delle masse: se è così eseguite il codice approntato finora e premete la caramella. Dovreste vedere il gatto muoversi in alto di dieci passi. Ora dovete impostare le altre direzioni del controller. Per creare il pulsante **DOWN** (giù), assicuratevi come prima che la caramella corrispondente sia collegata al pin 22 e a massa. Il codice di Scratch corrispondente dovrebbe essere questo:



› *ScratchGPIO7* ha la stessa interfaccia di Scratch 1.4 e permette anche ai più piccoli di realizzare programmi complessi

› I blocchi che consentono l'uso dei sensori forniscono un facile accesso ai pin del GPIO

```
Quando si clicca su bandiera verde
 per sempre
   se valore del sensore pin22 = 0
      cambia y di -10
      attendi 0.1 secondi
```

Bene, avete così creato i due pulsanti per muovere in su e in giù lo sprite modificandone la posizione lungo l'asse y. Ora dovete usare un codice simile per gli spostamenti verso destra e verso sinistra: vi basta sostituire il blocco **cambia y** con **cambia x**. Per muovere il gatto a sinistra userete un valore negativo, mentre per spostarlo a destra uno positivo. Ricordate poi che il pulsante per la sinistra è collegato al pin 24, mentre quello per la destra al 26. Ok, il codice del controller è finito. Adesso però dovete scrivere quello dell'ostacolo... vi sembra che ci possa essere un gatto senza un cane che lo insegue? Per il cane vi servono due sequenze di codice. La prima controlla i suoi movimenti e di base è un ciclo infinito che istruisce lo sprite a muoversi di cinque posizioni verso la posizione del gatto (Sprite1), per poi attendere 0,1 secondi.

```
Quando si clicca su bandiera verde
 per sempre
     punta verso Sprite1
     fai 5 passi
     attendi 0,1 secondi
```

La seconda sequenza, invece, serve a capire quando il cane raggiunge il gatto, cioè quando Sprite2 (il cane) tocca Sprite1. In quel caso il cane dice "Preso!" e continua l'inseguimento:

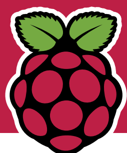
```
Quando si clicca su bandiera verde
 per sempre
     se sta toccando Sprite1
        dire Preso! per 2 secondi
        attendi 1 secondo
```

Bene, queste sono le basi del gioco, ma ci sono ampi spazi di miglioramento. Per esempio, provate da soli a capire come aggiungere uno sfondo, una musichetta o un effetto sonoro quando il cane raggiunge il gatto. Buon divertimento! LXP



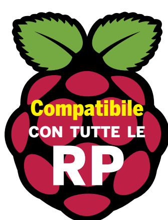
Tip

Potete usare anche della carta stagnola o altro materiale conduttore per costruire un'interfaccia di controllo. Potete, per esempio, realizzare un muro del suono collegando fogli di carta stagnola ai pin del GPIO e innescare il "rumore" con la mano indossando un guanto conduttore collegato a uno dei pin di massa della RP.



Facciamo un po' di musica!

La redazione di LXP vi regala un po' di musica pop usando solo la Raspberry Pi e Raspbian



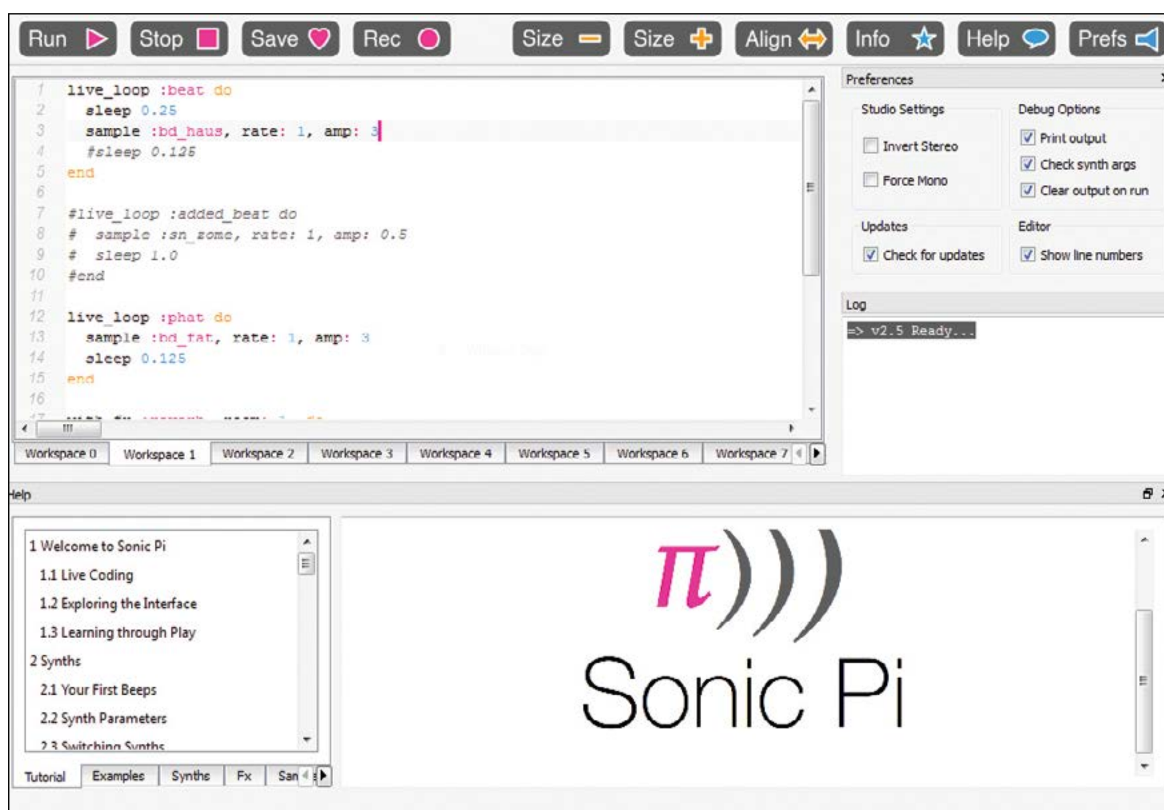
Iniziamo col dire che per portare a compimento questo progetto va bene qualunque modello di Raspberry Pi dotato di Raspbian. Nelle guide dei mesi scorsi ci siamo concentrati sulla creazione di progetti basati su vari componenti elettronici, con i quali era possibile interagire "fisicamente"; questa volta la nostra attenzione è sul mondo della musica basandoci su **Sonic Pi**. Lo scopo di Sonic Pi è di introdurre un po' di creatività nella programmazione usando la musica, cercando al contempo di ridurre le difficoltà che incontrano i bambini quando iniziano a programmare, per esempio quando devono capire il concetto di codice indentato e gli errori di sintassi. Sonic Pi usa il linguaggio di programmazione Ruby, creato a metà degli anni '90 da Yukihiro "Matz" Matsumoto con lo scopo di realizzare un'alternativa dalla sintassi facilitata ai linguaggi diffusi in quel periodo. Sonic Pi perfeziona il linguaggio Ruby grazie a funzioni pensate apposta per favorire l'apprendimento da parte dei più piccoli. Questa soluzione la trovate preinstallata in Raspbian, ma se non avete una versione aggiornata di questa distro vi conviene aggiornare il tutto aprendo un terminale e digitando

```
$ sudo apt-get update
$ sudo apt-get upgrade
```

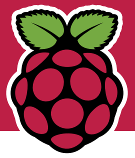
Se il sistema vi chiede una conferma prima di proseguire con l'installazione, rispondete di sì. Trovate Sonic Pi nel menu **Programmazione**; avviatelo e vedrete che la sua interfaccia è sostanzialmente divisa in tre pannelli orizzontali affiancati. In cima trovate una serie di pulsanti che servono a controllare la riproduzione della vostra composizione, per salvarla in un file e anche per registrare l'audio in un file Wav. Gli altri pulsanti servono a ridurre o aumentare la dimensione del testo in cui è scritto il codice,

Tip

La funzione di autoallineamento del codice di Sonic Pi è sensazionale: se mentre scrivete il testo vi appare confuso, vi basta cliccare su **Align** per essere sicuri che tutto sia dove dovrebbe essere.

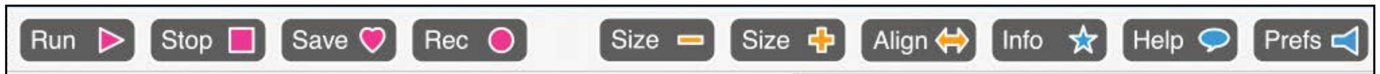


› **L'interfaccia di *Sonic Pi* è stata pensata per essere il più semplice possibile**



Programmazione live!

Il Dr. Sam Aaron (il creatore di Sonic Pi) è una persona brillante che conosce molto bene il proprio mestiere e ama dimostrarlo agli altri. Non stupisce, quindi, che sia un grande testimonial del *Live Coding*, la pratica di programmare di fronte a un pubblico. Sam fa parte della duo **Meta-ex** (**http://meta-ex.com**); l'altro componente è Jonathan Graham. Assieme mescolano la programmazione e gli strumenti musicali per realizzare performance uniche, con il pubblico che può osservare il codice mentre si trasforma per adattarsi ai toni e al ritmo della musica. A un recente evento, OpenTech, svoltosi a Londra, abbiamo visto l'uso di dati diffusi con licenza libera e registrati durante un disastro naturale, usati per dare forma alle note di un brano musicale particolarmente eclettico. In quel caso non si è trattato di Live Coding (il brano era stato preregistrato) ma grazie a Sonic Pi si potrebbero usare degli open data dal vivo in futuro!



› I pulsanti in cima all'interfaccia controllano la riproduzione e la registrazione, ma anche la dimensione del font del codice e il suo allineamento

allinearlo in automatico e accedere alla finestra di aiuto e di configurazione. Il pannello centrale contiene l'area per la scrittura del codice; quest'area è divisa in vari spazi di lavoro (*workspace*) che consentono di realizzare composizioni multiple. Alla destra del codice ci sono le preferenze, mentre nel pannello in basso trovate l'help, la documentazione e un gran numero di brani d'esempio. Iniziate la vostra avventura musicale suonando una prima nota. Nel pannello del codice scrivete

```
RUBY//play 60
```

Ora premete il pulsante **Run** e dovreste sentire molto brevemente il suono della nota. Ma cosa vuol dire quel 60? Questo valore si riferisce allo schema numerico usato dal protocollo MIDI (*Musical Instrument Digital Interface*), usato professionalmente per trasmettere le note musicali tra computer e strumenti digitali. Il 60 si riferisce alla nota C4 e potete usare direttamente questo nome al posto del valore numerico:

```
play :c4
```

Per suonare una serie di note, potete scrivere qualcosa di simile:

```
play :c4
sleep 1
play :g4
sleep 1
play :d4
sleep 1
```

Questo modo di procedere è corretto, anche se un po' lungo. Per limitare il numero di comandi da scrivere, Sonic Pi prevede la possibilità di indicare una sequenza di note all'interno di un'unica istruzione:

```
play_pattern [ :c4, :g4, :d4 ]
```

Forse le note sono un po' troppo lente per voi? Niente paura, Sonic Pi ha una soluzione anche per questo. Per velocizzare la riproduzione di un brano, potete impostare i BPM (*Beats per Minute*) velocizzando il suono prodotto dalla sequenza:

```
use_bpm 240
play_pattern [ :c4, :g4, :d4 ]
```

Usare i sintetizzatori

Bene, ora che siete in grado di riprodurre una serie di note è arrivato il momento di fare un passo in avanti e imparare l'uso dei synth (sintetizzatori). Un sintetizzatore consente di riprodurre una nota con strumenti diversi, così come fanno le tastiere o altri strumenti digitali. Quindi, modificate il codice di prima in modo che usi un synth:

```
use_synth :dsaw
 play :c4
```

Premete **Run** per capire la differenza. Note, strumenti... cosa manca? Un loop! Sonic Pi può creare loop infiniti usando il costrutto **loop do...end**. Tutto il codice presente al suo interno verrà ripetuto (suonato) per sempre. Per assicurarvi che il codice sia indentato in modo corretto, dopo averlo scritto, premete il pulsante **Align** presente nel pannello in alto:

```
loop do
   use_synth :dsaw
   play :c4
   sleep 1
end
```

Premete di nuovo **Run** e sentirete la nota c4 suonata una volta al secondo fino a quando non vi deciderete a premere il pulsante **Stop**. Ma se invece voleste eseguire il loop un numero ben definito di volte? Anche in questo caso Ruby ha una semplice soluzione:

```
2.times do
   use_synth :dsaw
   play :c4
   sleep 1
end
```

Esiste poi un altro tipo di loop, **live_loop**. Si tratta di un ciclo infinito utilizzabile per una performance di live coding. I cambiamenti fatti sul codice all'interno di un live_loop non ha effetto immediato, ma solo dopo che l'utente ha premuto il pulsante **Run**. Live_loop consente di creare codice concorrente facendo girare assieme più brani che formano, per esempio, un sottofondo musicale. La sintassi di un live_loop è simile a quella del loop normale, ma richiede l'assegnazione di un nome. Come esempio, le righe di codice seguenti creano un live_loop chiamato **beat** che incorpora la funzione **play_pattern** vista prima. Come al solito, premete **Run** per avviare l'esecuzione.

```
live_loop :beat do
   use_synth :dsaw
   play_pattern [ :c4, :g4, :d4 ]
   sleep 1
end
```

Ora cambiate **c4** con **f4** e premete di nuovo **Run**: dovreste sentire il cambiamento di tono della nota. Potete anche mandare al contrario la riproduzione usando la pratica funzione **.reverse** di Ruby:

```
live_loop :beat do
   use_synth :dsaw
```

Tip



All'interno di Sonic Pi potete trovare tutta la documentazione che vi serve per capire come funziona questo tool. I tutorial scritti da Sam coprono sia i concetti di programmazione che quelli musicali necessari alla creazione di un brano. Li trovate nel pannello inferiore dell'interfaccia.

»

```
  play_pattern [ :c4, :g4, :d4 ].reverse
  sleep 1
end
```

Oltre che suonare le note in avanti o all'indietro, è possibile farlo in modo casuale usando due funzioni:

```
play rrand(50, 100)
play rrand_i(50, 100)
```

Il primo **rrand** può suonare qualunque nota compresa tra 50 e 100, incluso qualunque valore MIDI in virgola mobile, mentre il secondo, **rrand_i**, riproduce solo valori MIDI interi compresi nello stesso range. Una buona pratica di programmazione prevede di raccogliere brani di codice da eseguire più volte all'interno di una funzione; ogni volta che sarà necessario eseguire quella serie di istruzioni, basterà richiamare questa funzione tramite il suo nome:

```
define :loopy do
  use_bpm 480
  use_synth :dsaw
  play_pattern [ :c3, :c4, :c5, :c6 ]
  sleep 0.5
end
live_loop :testy do
  loopy
end
```

In questo esempio si crea una funzione chiamata **loopy** e si usa il costrutto **do...end** per memorizzare il codice che imposta le battute per minuto a 480, manda in riproduzione un pattern usando il synth **dsaw** e, infine, mette tutto in pausa per mezzo secondo. Loopy viene poi richiamata usando il suo nome all'interno di un live_loop. L'ultimo concetto che dovete conoscere è quello dei campioni (*sample* in inglese). Si tratta di segmenti audio preregistrati, usati molto nell'industria della musica per abbellire dei brani usando clip tratte da canzoni classiche. Per usare un sample in un nuovo live_loop dovete inserire il codice seguente sotto il **live_loop :beat** esistente:

```
live_loop :samples do
  sample :loop_amen
  sleep sample_duration :loop_amen
end
```

L'istruzione **sleep** presente in quest'ultimo ciclo è inusuale: al posto del classico valore intero, o in virgola mobile, che determina la lunghezza della pausa, si dice a Sonic Pi di scoprire quanto dura il campione indicato e di usare questa lunghezza come valore per sleep.

Il primo brano completo

A questo punto avete tutte le conoscenze di base per riuscire a costruire il vostro primo componimento. Cliccate su un nuovo spazio di lavoro e iniziate il codice creando un

```
live_loop :beat do
  sleep 0.5
  sample :bd_boom, rate: 1, amp: 2
end

with_fx :reverb, room: 1  do
  live_loop :melody do
    use_synth :beep
    use_random_seed 10
    ns = (scale :g3, :major_pentatonic, num_octaves: 3)
    16.times do
      play ns.choose, detune: 6, release: 0.1, amp: 0.5,  cuto
      sleep 0.125
    end
  end
end
```

Workspace 0 | Workspace 1 | Workspace 2 | Workspace 3 | Workspace 4 | Workspace 5 | W

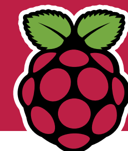
› L'interfaccia del programma vi mette a disposizione più spazi di lavoro che potete usare per provare loop musicali o brani diversi, ma anche per creare una composizione con più tracce da suonare in contemporanea

Coinvolgete anche Minecraft

Minecraft sulla Raspberry Pi è divenuto una *killer application* per insegnare il linguaggio Python nelle scuole e Sonic Pi ha raggiunto lo stesso status per quel che riguarda l'insegnamento musicale. Ma perché non fondere le sonorità di Sonic Pi con i mattoncini di Minecraft? A partire dalla release 2.5 di Sonic Pi potete farlo! Mentre scriviamo, questa caratteristica è ancora sperimentale, ma le funzionalità disponibili sono già numerose. Per esempio, è possibile ottenere la posizione del giocatore o di un blocco, cambiarla, e pubblicare dei dati nella finestra di chat. La sintassi è estremamente facile da assimilare e si integra perfettamente con quella standard di Sonic Pi. Ciò consente l'inclusione delle funzionalità di Minecraft in qualunque composizione già realizzata. Per usare Sonic Pi con Minecraft basta avviare quest'ultimo e caricare un mondo. Poi si esegue Sonic Pi e si sceglie una funzione di Minecraft per attivare la connessione, per esempio la finestra di chat:

```
mc_chat_post ("Ciao Mondo")
```

Eseguite il codice e il gioco è fatto.



live_loop chiamato beat. Esso conterrà il codice che andrà a formare il vostro brano. Per esempio potete inserire un sample e una breve pausa:

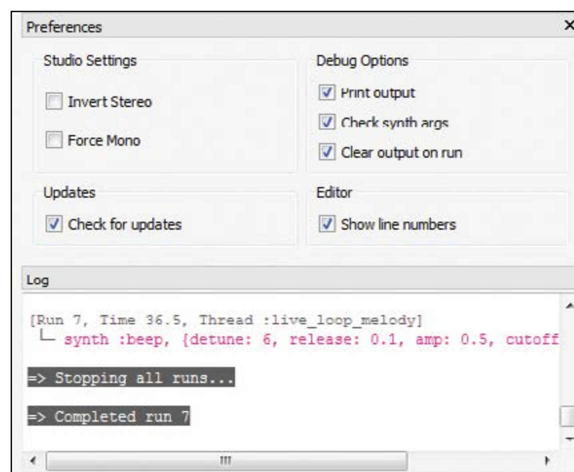
```
live_loop :beat do
   sample :bd_haus
   sleep 0.5
end
```

Avviate il codice e vedrete che il sample usato viene eseguito due volte al secondo (BPM pari a 120), quindi è molto veloce. Ora aggiungete un altro live_loop che contenga una melodia e preparatevi a nuovi effetti speciali...

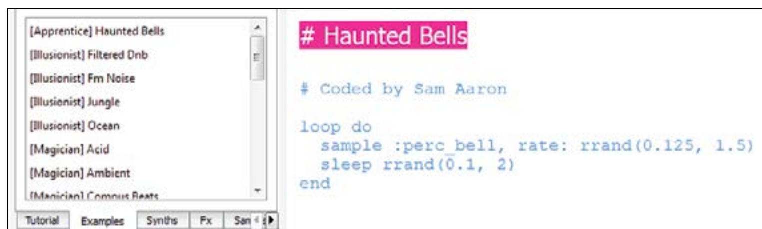
Usare i plug-in FX

Prima di iniziare questo nuovo ciclo ipotizzate di voler assegnare anche qualche effetto al suono; per effetto intendiamo cose come il riverbero, l'eco e la distorsione. In particolare prendete un riverbero: per inserirlo è necessario utilizzare il plug-in **fx** chiamato, guarda caso, **reverb**. Come argomento di reverb passate il valore **room 1**: si tratta della simulazione dell'effetto creato dalla "stanza" più grande disponibile in Sonic Pi. Dentro al nuovo loop, invece, provate a usare **beep** come synth. Per rendere le cose un po' più animate, infine, vi suggeriamo di far modificare a Sonic Pi l'inizio della composizione usando la funzione **use_random_seed**. Nell'esempio seguente noi abbiamo usato il numero 66678 come seme per il calcolo di un numero casuale (o semicasuale); provate altri valori per vedere come si modifica la melodia. A questo punto nel codice create una variabile chiamata **sound** al cui interno viene memorizzata la scala di note con chiave g3. Non è ancora finita. Adesso impostate un nuovo loop (da ripetere 16 volte) in cui si scelgono le note da suonare dalla variabile sound sulle quali si eseguono una serie di operazioni allo scopo di modificarne la timbrica: si stona leggermente la nota; si alterano dissolvenza e rilascio in modo che le note finiscano rapidamente; si dimezza il livello del volume delle note suonate. Infine si usa una nota casuale tra 70 e 80 per modificare certe frequenze d'uscita. Ecco il codice che realizza tutto ciò:

```
with_fx :reverb, room: 1 do
```



› L'area delle preferenze contiene tutto ciò che vi può servire per controllare l'output dell'audio e anche diverse opzioni per il debug e dell'editor. C'è anche il log che mostra l'output della vostra composizione musicale



› ***Sonic Pi*** **include tantissimi esempi già pronti; per usarli dovete copiarli e incollarli nell'area del codice. Provateli tutti per capire come ognuno dei sintetizzatori e dei plug-in FX può arricchire la vostra composizione audio**

```
  live_loop :melody do
    use_synth :beep
    use_random_seed 66678
    sound = (scale :g3, :major_pentatonic, num_octaves: 3)
    16.times do
      play sound.choose, detune: 6, release: 0.1, amp: 0.5,
 cutoff: rrand(70, 80)
      sleep 0.125
    end
  end
end
```

Premete **Run** per ascoltare il risultato. Ricordate che potete modificare il seme casuale per produrre un suono diverso. Potete anche modificare la scala usata, sostituendo g5 o c4, per esempio, alla scala g3 usata da noi. O ancora, è possibile trasformare **major_pentatonic**: in **minor_pentatonic** per produrre un suono più cupo e più triste. Manca però ancora qualcosa... un po' d'atmosfera. Nessun problema, basta crearne una. Aggiungete un nuovo live_loop, sempre usando qualche effetto. Questa volta vi suggeriamo di ricorrere al plug-in **ixi_techno** inserendo un filtro passa basso tra i valori minimi e massimi di cut off. Assegnate il nome **ambience** a questo nuovo loop e inserite un ciclo **do...end** da ripetere otto volte. Usando il *synth hollow*, dovete far suonare la nota c3 con un'ampiezza di 0,5. Ora serve una pausa di un secondo prima di far partire un altro ciclo, anch'esso da eseguire otto volte. Questa volta adoperate il synth **ambi_choir** alla velocità standard mixando però il suo suono in modo che appaia di sottofondo. Altra pausa di un secondo e poi chiudete tutti i cicli aperti. Ecco il codice:

```
with_fx :ixi_techno do
  live_loop :ambience do
  8.times do
    use_synth :hollow
    play :c3, amp: 0.5
    sleep 1
    8.times do
      sample :ambi_choir, rate: 1, amp: 0.2
      sleep 1
    end
  end
 end
end
```

Come al solito, fate partire la riproduzione. Non vi piace? Cambiarla è facile: avete tantissimi parametri su cui agire, quindi sperimentate senza timori. Tirando le somme, usando Sonic Pi e poche righe di codice siete riusciti a creare un loop sonoro che potete registrare premendo il pulsante **Record** per poi usarlo nei vostri filmati su YouTube o ovunque vogliate. Se siete un po' pigri e non volete digitare tutto il codice, potete scaricarlo dall'URL **http://bit.ly/LXP159_sonic**. LXP



Tip

Sonic Pi si basa sulla sperimentazione, quindi liberate la vostra mente da ogni condizionamento pregresso e scrivete semplicemente ciò che vi sembra naturale. La musica sgorgherà libera dal vostro codice!

L'angolo di Android

News, recensioni e guide sul sistema operativo libero per smartphone

Se hai news da segnalarci o dei commenti scrivici ad angolo_android@linuxpro.it



Ecco i nuovi Nexus!

Disponibili anche in Italia Nexus 5X e 6P insieme ad Android 6.0 Marshmallow

Come previsto, Google ha presentato i due nuovi Nexus di cui si sapeva praticamente già tutto grazie ai vari rumors e anticipazioni. Dopo il semi flop del Nexus 6 del 2014, che era veramente troppo grosso per essere definito uno smartphone, Google quest'anno ripropone il **Nexus 5** nella **versione X** potenziandolo come batteria e fotocamera mentre il **Nexus 6 versione P** pur avendo uno schermo da 5,7" diventa più piccolo e maneggevole. I due nuovi Nexus sono prodotti rispettivamente dalla coreana LG e per la prima volta dai cinesi di Huawei. Il nuovo Nexus 5X non presenta grandi novità rispetto al modello precedente presentato esattamente due anni fa: lo schermo è leggermente più grande (5,2'') ma la risoluzione Full HD e 424 ppi è rimasta la stessa. Il processore SnapDragon 808 è decisamente più potente ma la memoria RAM disponibile è rimasta di 2 GB. Più potente e sensibile la fotocamera ma senza lo stabilizzatore ottico dell'immagine, la porta USB usa il nuovo standard tipo C ma l'adattatore non è in dotazione. Unica vera novità interessante è il sensore di impronte digitali sistemato sul retro. Tutto questo al prezzo non proprio economico di 479 € per la versione da 16 GB, con un aumento di 130 € rispetto al prezzo del Nexus 5 di due anni fa e senza nemmeno la ricarica wireless. Decisamente più interessante allora il nuovo top di gamma Nexus 6P realizzato da Huawei che soffre solo il prezzo di 699 per la versione da 16 GB che arriva addirittura a 849 € nella versione da 128 GB, in pratica prezzi paragonabili ai top di gamma sia in campo Android che iOS. Rispetto al modello dell'anno scorso, Nexus 6P è decisamente più piccolo e maneggevole grazie alla struttura in alluminio con uno spessore di soli 7 millimetri e pur conservando un maxi display da 5,7 pollici con risoluzione QHD da 2.560x1.440 pixel. Per gestire uno schermo di questa risoluzione Google ha scelto il processore più potente oggi disponibile, lo SnapDragon 810 di Qualcomm, un octa-core da 2 GHZ che all'inizio aveva dato problemi di surriscaldamento ma che ora sembra funzionare al meglio. La RAM è da 3 GB e la memoria parte da 32 GB per arrivare ai 128 GB. La fotocamera principale utilizza un sensore Sony da 12,3 MXP con apertura f/2.0, ideale per gli scatti con poca luce e in grado di girare video a 4K, slow motion a 240 frame al secondo e scatti a raffica a 30 immagini al secondo. Sul retro c'è anche il sensore di impronta digitale **Nexus Imprint** compatibile con Android Pay. La batteria da 3400 mAH garantisce l'autonomia mentre la ricarica veloce può avvalersi della nuova porta USB Tipo C che troveremo prossimamente anche su molti PC portatili. LXP



› **I due nuovi *Nexus 5X* e *6P* saranno disponibili anche in Italia a partire da novembre**

Non solo smartphone

All'evento del 29 settembre di San Francisco, Google ha mostrato anche due nuovi Chromecast per lo streaming sul televisore, uno con il Wi-Fi potenziato e l'altro dedicato ai puristi dell'audio. **Chromecast 2015** e **Chromecast Audio** sono già disponibili a un prezzo di 39 €: il primo può vantare oltre al supporto alle reti Wi-Fi veloci anche un processore due volte e mezzo più veloce rispetto al modello precedente che dovrebbe trasferire senza problemi video alla massima definizione. Chromecast Audio è uguale al Chromecast 2 ma ha un'uscita jack al posto dell'HDMI, permette infatti il collegamento agli speaker del sistema di casa via cavo. LXP

Redmi Note 2

La categoria degli smartphone sotto i 200 euro si arricchisce di un nuovo modello

I produttori cinesi sono gli unici che in questo 2015 siano riusciti ad aumentare notevolmente le vendite grazie a un ottimo rapporto qualità/prezzo che marchi storici come Samsung e Sony non sono più in grado di raggiungere. Spicca sicuramente tra i produttori cinesi Xiaomi che, dopo avere iniziato realizzando ottimi cloni di iPhone sia sotto il punto di vista hardware sia software, ora ha imboccato un proprio stile vincente in tutto il mondo. Nonostante gli smartphone Xiaomi non siano ufficialmente importati in Italia è comunque possibile acquistarli online da importatori ufficiali senza dover nemmeno pagare le spese di dazio per l'importazione diretta.
Noi per esempio abbiamo trovato **Redmi Note 2** su Amazon Italia a 199 €, ma è possibile spendere ancora meno cercando in uno dei tanti siti online che lo importano direttamente.
Con il suo schermo IPS da 5,5 pollici, Redmi Note 2 può essere definito un phablet, categoria che oggi va decisamente per la maggiore. La struttura è quella classica in policarbonato molto solido che contraddistingue da sempre i dispositivi Xiaomi, con uno spessore di soli 8 millimetri e un peso ragionevole di 160 grammi.



È disponibile in due versioni: Redmi Note 2 standard con processore MediaTek Helio X10 MT6795 Cortex-A53 da 2 GHz con 2 GB di RAM e 16 GB di memoria e **Prime** che costa in media 40 euro in più e integra un precessore da 2,2 GHz e 32 GB di memoria interna. La memoria non è però un problema visto che esiste un alloggiamento per schede di memoria microSD sistemato sotto la batteria estraibile. Tra le connessioni disponibili oltre alle classiche Wi-Fi, Bluetooth e GPS non si può dimenticare quella a raggi infrarossi che insieme all'app **Remote** in dotazione permette di usare Redmi Note 2 come telecomando per la TV e gli altri dispositivi presenti in soggiorno. Manca invece la connessione NFC che però al momento è poco sfruttata. La sezione telefonica permette l'alloggiamento di due Micro SIM con una connettività LTE fino a 150 MBps. Occorre segnalare però il mancato supporto alla banda 800 MHz in 4G che con alcuni gestori potrebbe precludere la navigazione veloce.

Grande schermo

Nei modelli 'economici' lo schermo è spesso di scarsa qualità, fortunatamente non è questo il caso di Redmi Note 2 il cui pannello IPS da 5,5 pollici oltre a vantare una risoluzione Full HD è molto ben contrastato e visualizza colori naturali anche sotto la luce diretta del sole. Meno bene invece va la fotocamera che, pur avendo un sensore da ben 13 megapixel, è piuttosto lenta nella messa a fuoco e soffre le situazioni con poca luce. Per fortuna è possibile regolare manualmente ISO e bilanciamento del bianco riuscendo così a ottenere scatti di buona qualità.
Redmi Note 2 integra una batteria da ben 3,060 mAh, ma nonostante la capienza superiore alla media l'autonomia è appena sufficiente per arrivare a fine giornata. Con il Wi-Fi e il 3G attivo per tutto il giorno siamo riusciti a ottenere 4 ore di schermo acceso e 12 ore complessive. Abbassando l'uso dello schermo a 2 ore l'autonomia non supera comunque le 16 ore complessive. Si tratta di valori molto bassi che si spera possano migliorare visto che si tratta di un dispositivo nuovo che verrà sicuramente aggiornato dal punto di vista software. E poi naturalmente c'è il sistema operativo che è il Miui 7 basato su Android Lollipop 5.1 ma che si differenzia molto dall'esperienza originale di Google. Tutte le app infatti sono presenti direttamente nella Home con un'interfaccia che ricorda molto quella dell'iPhone. In più vi sono svariati widget che permettono di regolare le principali impostazioni direttamente dal desktop. Anche le principali app come browser, fotocamera e riproduttore multimediale sono personalizzate Miui e offrono un'ottima resa generale permettendo, per esempio, di gestire anche i video in 4K senza problemi. In definitiva ricordando sempre che ci troviamo di fronte a un dispositivo dal prezzo inferiore ai 200 € non possiamo che dirci soddisfatti. L'unico vero limite rimane quello dell'autonomia che confidiamo però possa migliorare in uno dei prossimi aggiornamenti software. LXP



› ***Redmi Note 2*** **è disponibile in quattro colori e il retro può essere sostituito**

LINUX PRO Giudizio

Redmi Note 2

Produttore: Xiaomi
Web: www.mi.com
Prezzo: 199 € (amazon.it)

Caratteristiche	8.5
Autonomia	7
Prestazioni	8
Qualità/prezzo	9

» *Ottimo rapporto qualità/prezzo per un phablet che se la gioca con modelli costosi.*

Il voto di Linux Pro 8

Recensioni

Tutte le novità in campo software e hardware testate e valutate ogni mese dai nostri laboratori

Se vuoi segnalarci qualche novità scrivi a recensioni@linuxpro.it

Una breve legenda

Ogni test di questa sezione è accompagnato da un giudizio che riassume con quattro indici numerici le principali qualità dell'applicazione o del prodotto hardware messo alla prova. I laboratori di Linux Pro assegnano un voto da 1 a 10 alle seguenti categorie:

Caratteristiche: fornisce tutte le funzioni di cui abbiamo bisogno? È innovativo?
Prestazioni: esegue in maniera efficiente le sue funzioni? È veloce e affidabile?
Facilità d'uso: dispone di un'interfaccia grafica chiara e facilmente fruibile? La documentazione che lo accompagna è sufficientemente completa ed esaustiva?
Qualità/prezzo: ha un prezzo competitivo? Vale i soldi richiesti per il suo acquisto?

Il nostro giudizio viene poi riassunto da un voto finale, espresso anche graficamente. Ecco la legenda dei voti:

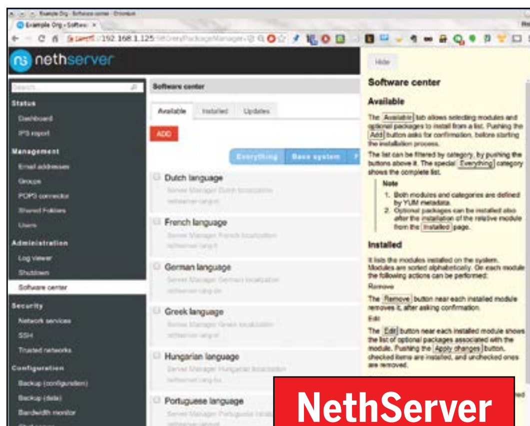
10 Nulla da eccepire. Un prodotto praticamente perfetto.
8-9 Un buon prodotto. I pochi difetti presenti non sono gravi.
6-7 Compie il suo lavoro ma necessita di ulteriori sviluppi.
5-4 Deve migliorare prima di raggiungere un voto sufficiente.
1-3 Un completo disastro. Gli sviluppatori devono tornare alla fase di progettazione.

Ricordiamo infine che i software citati nelle sezioni Confronto e Da non perdere sono spesso presenti nel DVD sotto la voce "Rivista" sotto forma di codice sorgente o binario.

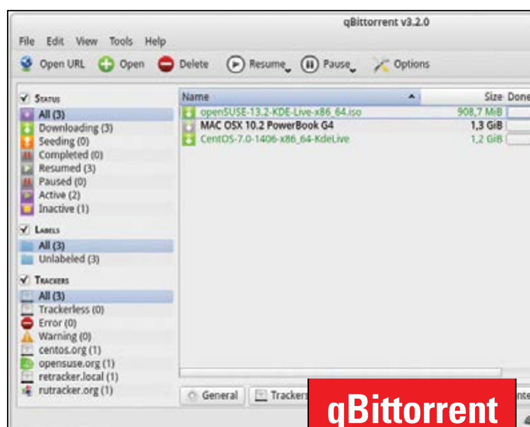
QUESTO MESE...



Acer Chromebook 15 C910 54M1



NethServer



qBittorrent

Acer Chromebook 15 C910 54M1

È effettivamente il Chromebook più grosso che abbiamo mai visto... ma le dimensioni sono davvero tutto?

In breve

» **CPU:** Intel Core i5-5200U Dual-core a 2,2 GHz
» **Grafica:** Intel HD Graphics 5500
» **RAM:** SDRAM DDR3L da 4 GB
» **Schermo:** 15,6 pollici full HD (risoluzione 1.920 x 1.080)
» **Capacità:** unità a stato solido da 32 GB
» **Porte:** HDMI, 1 USB 2.0, 1 USB 3.0
» **Connessioni:** Wi-Fi 802.11a/b/g/n/ac
» **Fotocamera:** 720p HD
» **Dimensioni::** 25 x 383 x 256 mm, 2,19 Kg

Rispetto alla maggior parte degli altri portatili con sistema ChromeOS, l'**Acer Chromebook 15** (**C910 54M1**) ha uno schermo più grande, una maggiore potenza di elaborazione e un prezzo più elevato. Pensato specificamente per studenti e insegnanti, grazie alla sua struttura robusta e alle sue immagini favolose, l'Acer è adatto a qualunque utente che possa permettersi di sborsare qualche euro in più e di occupare qualche centimetro in più. La prima cosa che si nota di questo modello sono le sue grosse dimensioni, che trovano riscontro in un peso di ben 2,19 Kg. È insomma un notebook bello grosso, poco adatto a chi si sposta continuamente. D'altro canto, è perfetto per studenti e insegnanti, che trascorrono lunghi periodi di tempo seduti davanti a un banco. Con il suo elegantissimo telaio nero dalla superficie tipo tessuto con finitura lucida, il C910 può dare un istantaneo tocco di classe a qualsiasi scrivania. Lo schermo da 15,6 pollici full HD con risoluzione 1.920 x 1.080 di questo portatile è semplicemente delizioso. Potrete guardare film, navigare in Rete e divertirvi con i videogiochi per lunghi periodi di tempo, senza affaticare troppo gli occhi. Chi ama guardare i film in compagnia apprezzerà l'ampio angolo visivo dello schermo, che consente a tre persone di collocarsi a un metro e mezzo di distanza senza percepire alcun alone d'ombra intorno ai bordi. Acer ha riservato circa tre dei 15,6 pollici di larghezza del piano della tastiera agli altoparlanti frontali del laptop, adeguati anche se non eccezionali. Il touchpad, che scricchiola un po', occupa altri tre pollici, mentre la tastiera ne occupa poco più di cinque. Questo rende la digitazione un po' scomoda, specie per chi ha dita piccole e poca esperienza nel battere sui tasti (i bambini, per esempio). Acer avrebbe fatto meglio a estendere la tastiera e i tasti, riducendo la larghezza degli altoparlanti e dividendo in due il touchpad.



› L'Acer trae grande beneficio dal suo schermo HD

Una CPU di classe

Acer ha creato un certo putiferio quando ha impiegato per la prima volta processori Broadwell, in quanto questi ultimi sono più veloci e più efficienti in termini di consumo energetico rispetto ai processori Celeron tipicamente utilizzati dai Chromebook. Questa scelta ha sostanzialmente trasformato i Chromebook come classe da dispositivi adatti solo a usi leggeri in notebook che si prestano all'impiego nel lavoro. Rispetto al Pixel 2, il C910 è un notebook leggermente superiore in termini di prestazioni, malgrado il suo prezzo sia inferiore. Nel test Octane, che misura le prestazioni JavaScript, il Pixel 2 ha totalizzato 24.564 punti, contro i 25.240 del C910. Per chi è interessato a questo genere di test, precisiamo che il notebook ha ottenuto un punteggio di 1301,5 in Mozilla Kraken e 192,5 punti in SunSpider. Il campione si è inoltre prodotto in ben 8 ore e 48 minuti di riproduzione video continuata con volume e luminosità dello schermo regolati sul 50%. Non saranno le 10 ore promesse da Acer... ma ci vanno vicino. In questo campo, il vantaggio del C910 rispetto al Pixel 2 è più ridotto: dal secondo modello siamo infatti riusciti a spremere 8 ore e 22 minuti di autonomia in un test più estremo. LXP



› Meno trasportabile di altri Chromebook... ma più potente

LINUX PRO Giudizio

Acer Chromebook 15 C910

Sviluppatore: Acer
Web: www.acer.co.uk
Prezzo: 512 €

Caratteristiche	7/10
Prestazioni	10/10
Facilità d'uso	9/10
Documentazione	8/10

» *L'Acer Chromebook 15 C910 è un prodotto potente e durevole che predilige le prestazioni alle dimensioni.*

Il voto di Linux Pro **9**

CrunchBang++ 1.0

Ripensando alle ore trascorse programmando, esaminiamo **CrunchBang++** per verificare se ha abbastanza ‘più’ da giustificare il suffisso...

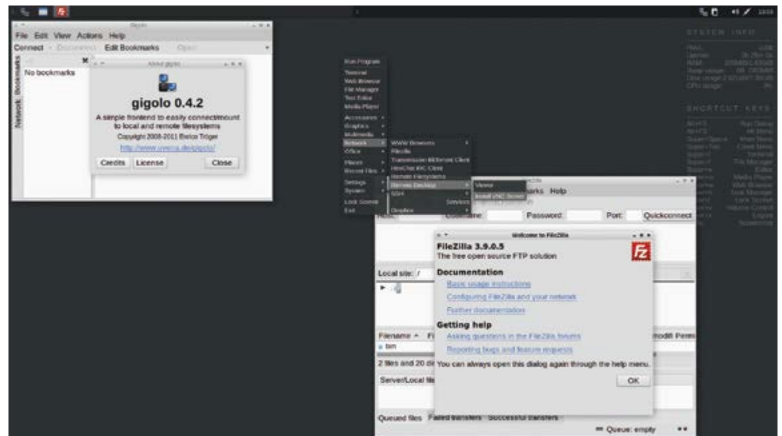
In breve

» Nota sia come CBPP sia come #!++ e basata su Debian 8, questa distribuzione utilizza il gestore delle finestre leggero OpenBox e offre un buon compromesso tra velocità e funzionalità. Comprende inoltre applicazioni di successo come VLC e Gimp, accanto ad alternative ‘leggere’ per molte altre applicazioni. Vedi anche: Puppy Linux, Porteus, Slitaz.

I progetti di Linux vanno e vengono. Sono molte le distribuzioni che, pur sostenute da vivaci comunità di utenti, sono state sospese per una ragione o per l’altra. L’annuncio della fine di CrunchBang, tuttavia, ha creato un certo shock. L’incredulità e il desiderio di portare avanti comunque il progetto hanno fatto sorgere un successore, denominato Bunsen Labs. Sebbene non abbia alcun collegamento con i precedenti sviluppatori di CrunchBang, CrunchBang++ (#!++) rappresenta un tentativo di proseguire la tradizione di una delle distribuzioni minimaliste più amate. #!++ si basa su Debian 8 e utilizza anche il suo programma di installazione. Se questa notizia avrebbe potuto preoccupare gli utenti qualche anno fa, nelle release più recenti il programma di installazione di Debian è divenuto sempre più accessibile. La distribuzione, tuttavia, non è in grado di avviare un sistema dal vivo.

Una marcia in più?

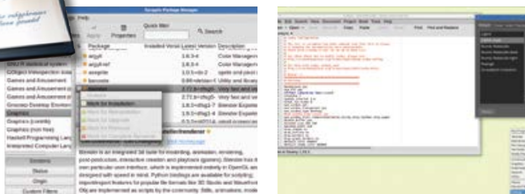
Una volta installata, la distribuzione vi presenta un desktop minimalista basato su OpenBox e avvia automaticamente uno script di configurazione post-installazione che vi guida attraverso una serie di opzioni di configurazione, quali l’installazione del supporto per la stampante, di Java Runtime Environment, di LibreOffice, della piattaforma LAMP, eccetera. Non siete obbligati ad avviare questo script in 12 punti, dato che richiede una connessione a Internet e potete lanciarlo in qualunque momento dal terminale utilizzando il comando `$ bpp-welcome`. La distribuzione offre uno script di installazione per importanti applicazioni, il che vi risparmierà la fatica di installarle manualmente tramite il gestore dei pacchetti Synaptic o lo strumento `apt-get`. Incredibilmente, la distribuzione offre programmi di installazione analoghi anche per DropBox e per il server VNC. Se siete abituati a una delle distribuzioni più moderne e sgargianti, troverete un po’ deprimente lo stile grigio ed essenziale di questa. D’altra parte, OpenBox è molto configurabile; volendo, potete modificare praticamente ogni elemento del desktop per renderlo più vivace. Oltre a riportare le statistiche di sistema essenziali, il monitor di sistema Conky, collocato sul lato destro del desktop, elenca anche un certo numero di tasti di scelta rapida predefiniti, per esempio quelli per l’avvio del file manager, del terminale, del lettore multimediale e via dicendo. Trascorrete molto tempo nella riga di comando e scoprirete che comandi come ifconfig non possono essere utilizzati da parte di un utente non root. La ragione è che /sbin non è incluso in $PATH. Avviando export PATH=$PATH:/sbin potrete risolvere il problema. Molte distribuzioni leggere rimpiazzano KDE o Gnome con Xfce o LXDE. Molte altre rinunciano del tutto all’ambiente desktop e creano l’interfaccia grafica tramite un gestore delle finestre come OpenBox. Se è vero che #!++ rientra in questa seconda categoria, questa distribuzione ha però fatto molto di più per dare vita a un sistema veloce a basso consumo di risorse. LXP



› **Malgrado il suo aspetto spartano, *CrunchBang++* è una distribuzione perfettamente attrezzata che può essere trasformata facilmente in un desktop altamente funzionale o utilizzata per un server LAMP**



Caratteristiche



Basato su Debian

Essendo basato su Debian, #!++ è straordinariamente stabile. Inoltre è velocissimo e può essere facilmente adattato a ogni utilizzo.

Ultra-configurabile

#!++ utilizza il versatilissimo OpenBox e comprende inoltre strumenti come il monitor di sistema Conky e Thunar FM.

LINUX PRO Giudizio

CrunchBang++ 1.0

Sviluppatore: Ben Young
Web: www.crunchbangplusplus.org
Licenze: Varie

Caratteristiche	8/10
Prestazioni	9/10
Facilità d’uso	9/10
Documentazione	5/10

» *CrunchBang++ è una distribuzione che non possiede un forum proprio ma mantiene tutto ciò che promette.*

Il voto di Linux Pro **8**

Kerbal Space Program v1.0

Andiamo coraggiosamente dove nessun uomo è mai giunto prima...

Specifiche

» Sistema operativo: Ubuntu 14.04 (supportata la maggior parte delle distribuzioni)

» CPU: Intel Core i5

» RAM: 4 GB

» Spazio su disco: 2 GB

» Grafica: Qualunque scheda con Shader Model 4, 1 GB di VRAM

In **Kerbal Space Program** (KSP) si tratta di costruire razzi e pilotarli nello spazio. È probabile che lo sappiate già, dato che questo titolo è stato pubblicato per la prima volta in versione alfa nel lontano 2011. L'ultima versione introduce alcuni progressi degni di nota, benché nessuna delle aggiunte alteri l'azione di gioco in misura significativa. D'altronde, le fondamenta di Kerbal Space Program sono state gettate da tempo. Gli aggiornamenti avevano la funzione di ampliarle e nel corso degli anni hanno finito per dare forma al gioco così come lo conosciamo oggi. Ci sono nuovi tipi di missioni, più componenti per le astronavi, astronaute Kerbal femmine, effetti grafici rinnovati e un nuovo pannello **Engineer Report** che vi avverte degli errori più grossolani commessi durante la costruzione dell'astronave.
La modalità **Sandbox** si può considerare l'essenza di Kerbal Space Program. Questa modalità vi offre risorse illimitate e pieno accesso ai numerosi componenti offerti dal gioco. Le partite in modalità Sandbox sono per definizione prive di struttura.
Al tempo stesso, esiste una progressione naturale che potete decidere di seguire. Prima c'è una Luna, poi ci sono i pianeti, ognuno dei quali è abbastanza lontano da rendere sempre più difficile raggiungere il successivo. Se invece preferite missioni più strutturate, c'è anche una modalità **Career**. All'inizio le vostre possibilità in termini di costruzione sono estremamente limitate. D'altro canto, iniziare la partita risulta un po' più facile e veloce. Avendo a disposizione un numero minore di componenti per le astronavi, avete tempo per capire a che cosa serve ciascuno di essi.
Le diverse tipologie di contratto possono creare scenari alquanto impegnativi anche per i conoscitori di questo gioco.



› **Non dovremo mica rimanere in questa posizione per tutti i 10.000 metri del decollo?**

Una simulazione con un'anima

Come ogni simulazione, Kerbal Space Program correva il rischio di risultare freddo e arido. Il fatto che non lo sia per niente è dovuto alla sua presentazione e alla natura degli stessi Kerbal. Sono goffi e pasticcioni nel lavoro e nel gioco vengono classificati in funzione del loro coraggio e della loro stupidità. Sono i Kerbal a riportare la fisica con i piedi per terra. In un gioco più serio, errori e fallimenti creerebbero un senso di tragedia oppure, peggio ancora, apparirebbero puramente teorici. I Kerbal introducono nella simulazione una dimensione umana e anche un elemento farsesco. Quando uno di loro muore vi dispiace... ma neanche troppo.
È facile scambiare Kerbal Space Program per un gioco incentrato sulla tecnica e sulla progettazione. Spesso lo è, per esempio quando si tratta di regolare il centro di gravità di un'astronave, di accrescerne la stabilità o di usare la mappa orbitale per effettuare un lancio d'emergenza controllato verso una destinazione lontana. Esteticamente non è un gioco particolarmente eccezionale. Le texture dei pianeti sono elementari e a bassa risoluzione. La grafica è funzionale, cioè proprio come dovrebbe essere. Occasionalmente, però, vi ritroverete a galleggiare nello spazio proprio mentre il sole emerge da dietro Kerbin, oppure scorgerete il lontano bagliore di un altro pianeta. Ed è appunto a distanza che il motore di gioco dà il meglio di sé quanto a illuminazione. In movimento, Kerbal Space Program è perfettamente in grado di trasmettere la maestosità e il senso di stupore che caratterizzano l'esplorazione dello spazio. È una sorta di promemoria: lo abbiamo fatto davvero. Spinti dalla scienza, dal desiderio di scoprire, nonché da una competizione quasi apocalittica tra due superpotenze, abbiamo ficcato delle persone dentro a colossali macchine alimentate da combustibili e le abbiamo sparate fuori dal nostro pianeta. Lo abbiamo fatto... ed è stato splendido. Questo è un gioco raro e meraviglioso. LXP

LINUX PRO Giudizio

Kerbal Space Program

Produttore: Squad
Web: https://kerbalspaceprogram.com
Prezzo: 40,60 €

Giocabilità	10/10
Grafica	7/10
Longevità	10/10
Qualità/Prezzo	8/10

» *Un cocktail perfetto di scienza e ironia, che offre una ricca dose di avvincenti possibilità. Eccezionale.*

Il voto di Linux Pro **9**

Ultimaker 2

Abbiamo provato un dispositivo per la stampa 3D silenzioso e ad alta risoluzione



› **Una soluzione completa per la stampa 3D adatta alla casa, all'ufficio e alla scuola**

In breve

» Un'evoluzione del modello originale che offre un funzionamento molto meno rumoroso e ottimi risultati. Comprende inoltre il software Cura per il caricamento dei modelli, ottimamente progettato e intuitivo. Vedi anche: RepRap, Lulzbot Mini.

La **Ultimaker 2** utilizza componenti e software Open Source ma è un prodotto decisamente compiuto, disponibile solo in versione pre-assemblata. Le versioni sono tre: **Ultimaker 2 Go**, **Ultimaker 2** e **Ultimaker 2 Extended**. I lati in legno tagliato al laser e il pannello di controllo che conferivano alla Ultimaker il suo classico stile 'fatto in casa' sono ormai un ricordo; i nuovi materiali utilizzati per l'esterno del dispositivo sono più rifiniti e robusti. L'area di stampa è un fattore importante e la Ultimaker 2 offre un piano riscaldato da ben 23 x 22,5 x 20,5 cm. I modelli tridimensionali possono essere caricati tramite l'ingresso per scheda SD collocato sulla parte anteriore; per selezionare il modello da stampare si usano il display LCD e la manopola, anch'essi alloggiati nella parte anteriore. I modelli vengono caricati sulla scheda insieme alle impostazioni di qualità per mezzo del software **Ultimaker Cura**. Sul retro del dispositivo ci sono l'alloggiamento per il filamento, l'interruttore di accensione e una porta USB che va utilizzata per l'aggiornamento del firmware, più che per la stampa in tethering. Basta estrarre la Ultimaker dalla confezione per capire che, a parte la necessità di collegare l'alloggiamento per il filamento sul retro e rimuovere l'imballaggio, la stampante è praticamente pronta per l'uso. L'alloggiamento va inserito fino a quando si sente uno scatto; è poi sufficiente usare il pannello di controllo sulla parte anteriore per selezionare il materiale e inserire il filamento nell'alimentatore sul retro. Durante tutta questa procedura, le istruzioni sul display LCD vi dicono esattamente che cosa fare; dopo un paio di minuti, il filamento viene caricato ed è pronto per l'uso. A quel punto è sufficiente preparare il piano riscaldato mediante colla stick, inserire la scheda SD, cliccare su **Print**, selezionare uno dei modelli pre-caricati e la procedura di stampa avrà inizio.

Una stampante irresistibile

Quando sarete pronti per stampare dei modelli con la Ultimaker 2, dovrete per prima cosa caricare il vostro modello nel software **Cura**. Attualmente Ultimaker si occupa della manutenzione di questa applicazione, che tuttavia viene utilizzata anche da altre case produttrici quali Lulzbot ed è straordinariamente ben progettata e intuitiva. Permette di regolare dimensioni, angolo di rotazione e scala, tutti parametri che sono riportati sulla piastra di stampa virtuale; inoltre, il software consente di regolare le impostazioni di qualità di stampa. La Ultimaker 2 è poco rumorosa: potrete senz'altro lasciarla in funzione senza disturbare le altre persone presenti in ufficio o in casa o i vicini. Il rumore è un elemento da tenere in considerazione, specie per chi, dovendo lavorare a breve distanza dalla stampante, rischia di esserne distratto. D'altronde, la possibilità di caricare i file tramite scheda SD vi permette di lasciare tranquillamente la stampante in un'altra stanza, il che rappresenta un grosso vantaggio rispetto alle stampanti che devono rimanere collegate al computer per funzionare. I risultati dei nostri test spaziano da un'ora con l'impostazione **Fast Print** a 10 ore con l'impostazione Ulti per la stampa di una scatola del cambio: sono tempi ottimi, paragonabili a quelli di altre stampanti come la Lulzbot Mini. La qualità di stampa, con l'impostazione massima di 20 micron, è eccellente e sebbene si possano notare gli strati, basta ripulire un po' il modello per vederlo trasformarsi. La Ultimaker 2 porta la stampa 3D a un nuovo livello. È un prodotto completo in termini di struttura, funzioni e software. Se siete alla ricerca di una stampante domestica, troverete molto interessante questo modello poco ingombrante e dall'area di stampa spaziosa; e il fatto che sia poco rumorosa la rende ideale per il lavoro e per la scuola. LXP

Caratteristiche



Alloggiamento per il filamento

Il filamento avvolto sul rocchetto sistemato sul retro garantisce un'alimentazione continua.



Poco rumorosa

La riduzione del rumore rappresenta un notevole progresso di questo modello.

LINUX PRO Giudizio

Ultimaker 2

Produttore: Ultimaker
Web: https://ultimaker.com
Prezzo: 2.000 €

Caratteristiche	9/10
Prestazioni	8/10
Facilità d'uso	9/10
Qualità/Prezzo	6/10

» *È facile da usare, offre stampe di qualità, supporta ABS e PLA ed è poco rumorosa.*

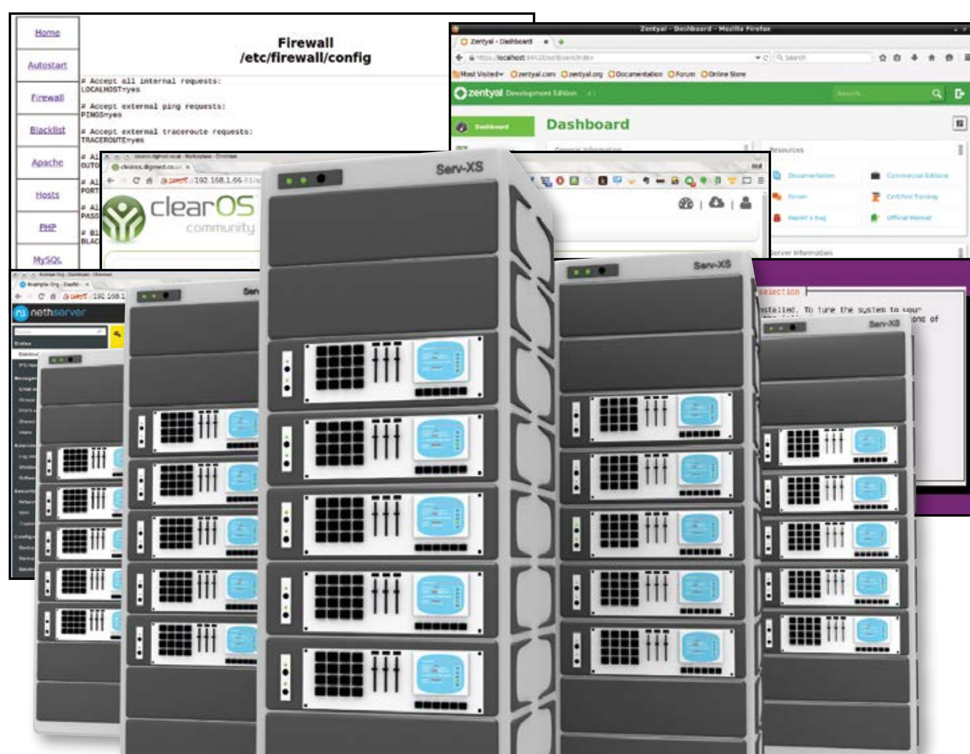
Il voto di Linux Pro **8**

Il confronto

Ogni mese mettiamo a confronto prodotti e programmi per farvi scegliere al meglio!

Distro per server

Il nostro articolo di copertina vi ha dato un assaggio delle distribuzioni per server, ora è il momento di analizzarle nel dettaglio...



Modalità del test

Le distro sono state installate in identiche macchine virtuali Qemu/KVM, così da permettere un migliore confronto operativo. In più, le abbiamo provate su hardware reale. Se però avete bisogno di un server commerciale, è meglio puntare su soluzioni chiavi in mano con tanto di supporto tecnico dedicato. Il nostro confronto è incentrato su un utilizzo domestico o in alternativa in piccoli uffici. Ecco perché abbiamo puntato molto sulla semplicità d'installazione e configurazione, nonché sulla flessibilità delle funzioni più rilevanti. Ciò non significa che sia impossibile usare queste piattaforme per progetti più importanti o creare un server partendo da una semplice distro Debian. Il nostro intento è però fornirvi alternative pratiche da utilizzare per attività non necessariamente aziendali.

Linux è da sempre una delle soluzioni più utilizzate in ambito server. Se però cercate una distro capace di creare una piattaforma domestica, quale dovreste scegliere? È proprio a questa domanda che vogliamo dare una risposta con il nostro confronto. In teoria, qualsiasi distribuzione può essere utilizzata come base per la costruzione di un server. Ci sono però alcune piattaforme più adatte di altre. Infatti, in caso si scelga una distro basilare, si dovranno aggiungere una quantità considerevole di componenti e funzioni. Per evitare di perdere tempo in questa attività, per fortuna esistono sistemi appositamente realizzati per fornirvi tutto quello di cui avete bisogno. Se state cercando una soluzione commerciale da usare in un'azienda di grosse dimensioni, consigliamo comunque di rivolgersi a professionisti del settore. Vi forniranno installazione e assistenza on-site e non dovrete preoccuparvi di niente. Se però volete sfruttare qualcosa da utilizzare in ambito privato o in un piccolo ufficio, le distro che vi presentiamo hanno tutte punti forti e deboli da valutare. Vediamo quindi qual è la migliore secondo le varie esigenze che potete trovarvi ad affrontare.

> "Per realizzare un server domestico ci sono soluzioni che forniscono tutto ciò di cui avete bisogno"

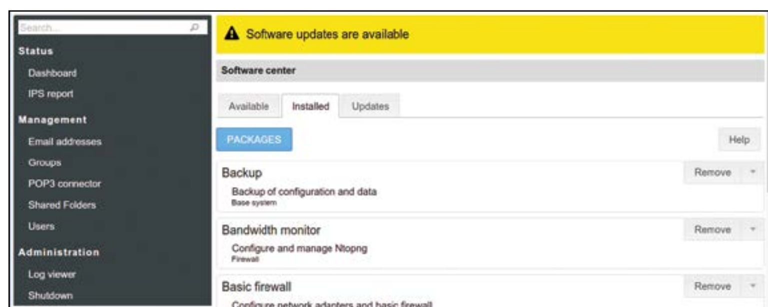
La nostra selezione

- ClearOS
- NethServer
- TheSSS
- Ubuntu Server
- Zentyal

Installazione

Sono facili da montare?

Tutte e cinque le distro prese in considerazione sono pensate per essere installate ed eseguite in un disco fisso. Gli installer sono gli stessi che utilizzate nella maggior parte delle piattaforme desktop (di solito in versione solo testo). In linea di principio, non ci sono molte opzioni da selezionare in fase d'installazione. **ClearOS** si basa su CentOS e utilizza l'installer grafico di Red Hat **Anaconda**. **NethServer**, invece, fornisce un'opzione per l'installazione automatica che consente di montare la distro sul disco con scelte predefinite. Non usa però un'interfaccia grafica che permetta di mettere mano al mouse. Per selezionare le varie voci, dovrete quindi muovervi con le frecce della tastiera. **TheSSS** (acronimo di *The Smallest Server Suite*) si presenta in modo leggermente diverso. Infatti, si avvia tramite un account di accesso che consente poi di gestire un server e il relativo programma d'installazione. Anche in questo caso, comunque, avrete a disposizione opzioni di configurazione minime. I partizionatori che trovate con i vari programmi d'installazione hanno caratteristiche piuttosto basilari. Quelli di ClearOS e NethServer, sotto questo profilo, sono addirittura i peggiori. Pur utilizzando **LVM**, riempiono il gruppo dei volumi con un unico elemento logico, vanificando tutti i benefici dello standard del gestore logico. **Ubuntu Server**, invece, è un passo avanti. Usa sempre LVM, ma consente di aggiungere i vari volumi logici assegnandogli i rispettivi punti di montaggio. Un aspetto, questo, molto importante in un server dove si vuole tenere separato il sistema operativo dal contenuto. A vantaggio di NethServer possiamo citare la sua procedura d'installazione automatica. Essa consente di avere la piattaforma pronta all'uso in pochi minuti. Al contrario, ClearOS, Ubuntu e **Zentyal** offrono scelte multiple. In definitiva, dipende molto da come si preferisce lavorare. L'approccio di NethServer è però più indicato se si ritiene utile installare più server.



› **Qui potete vedere *NethServer* che viene installato senza alcun input da parte nostra**

Verdetto

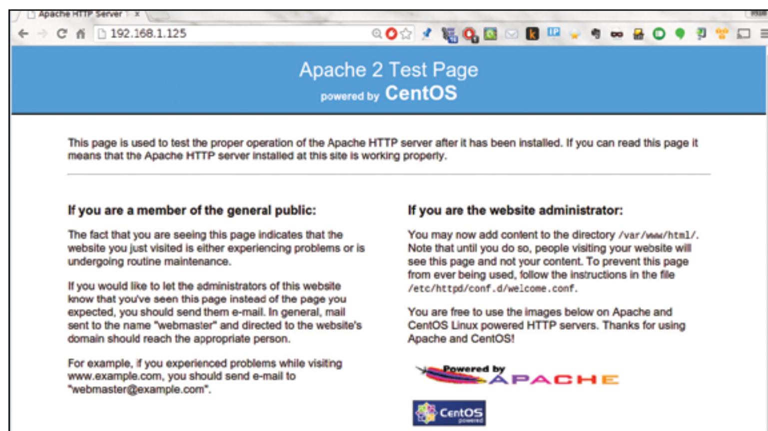
ClearOS ★★★★☆
NethServer ★★★★☆
TheSSS ★★★☆☆
Ubuntu Server ★★★★☆
Zentyal ★★★★☆

» *NethServer vince per merito della sua installazione automatica.*

I servizi più utili

Nessun server è uguale all'altro

La parola server ha un significato molto ampio. Nella maggior parte dei casi, si tende a definirlo come una piattaforma che consente di gestire pagine Web, email e file. Questo è proprio quello che fanno tutte le distro prese in considerazione. C'è però un'eccezione. Zentyal, infatti, non fornisce alcun servizio Web o FTP, bensì è incentrato sulla gestione dei file nei piccoli e medi uffici. Si basa su Ubuntu e permette di installare facilmente Apache o un qualsiasi altro componente simile che però dovrà essere configurato per l'occasione. TheSSS si rivela una distro estremamente leggera. Offre pertanto servizi indispensabili: Web server, Apache e supporto FTP, ma non per NFS o Samba. La mancanza più evidente riguarda però la gestione della posta. Gli altri, vale a dire ClearOS, NethServer e Ubuntu Server, utilizzano più o meglio gli stessi software per i seguenti servizi: Apache, Postfix e Dovecot (cyrus-imapd su ClearOS). Le principali differenze tra queste piattaforme si riducono quindi alla semplicità d'uso. Una menzione particolare la merita NethServer che consente di amministrare un gran numero di account di posta in modo facile e flessibile.



› **La maggior parte delle distro supporta Apache, ma dovrete comunque installarlo**

Verdetto

ClearOS ★★★★★
NethServer ★★★★★
TheSSS ★★★☆☆
Ubuntu Server ★★★★★
Zentyal ★★★★☆

» *ClearOS e NethServer hanno accesso a molti server CentOS.*

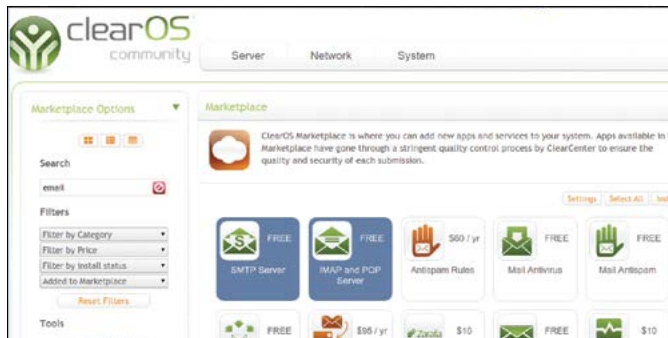
Amministrazione Web

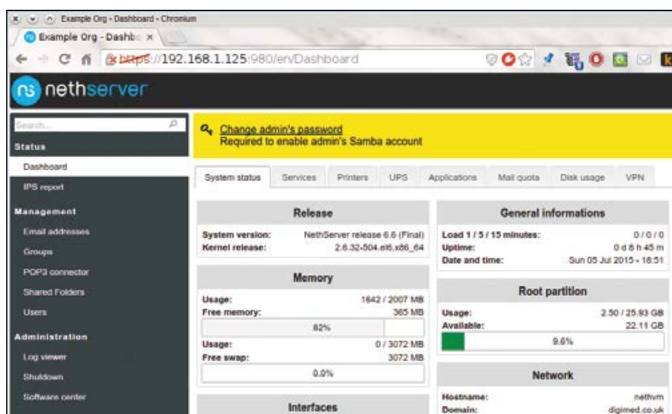
Molti server hanno bisogno di un accesso remoto per essere configurati

Gli amministratori di sistema più esperti fanno tutto tramite un terminale equipaggiato con Emacs. I comuni mortali, invece, preferiscono senza dubbio un'interfaccia grafica, in particolare per tutte le attività più complesse. A tal proposito, una buona distro server deve comunque disporre di un ambiente grafico capace di permettere una gestione oculata e veloce delle varie funzioni. Nella maggior parte dei casi l'interfaccia viene quindi fornita tramite browser. Così facendo, è possibile amministrare il tutto da qualsiasi luogo. Naturalmente questo non è l'unico sistema possibile per interagire con il server. Per fare le cose velocemente e soprattutto in modo sicuro, la migliore soluzione rimane una Shell con accesso SSH.

ClearOS ★★★★★

ClearOS dispone di un display grafico che mostra tutti i dettagli necessari al collegamento con la piattaforma server, nonché un link per modificare le impostazioni di rete. Quando vi collegate all'interfaccia Web, avrete a disposizione una procedura guidata utile all'installazione. Anche gli utenti meno esperti si sentiranno subito a proprio agio. Non c'è poi bisogno di registrare alcun account. Nell'ultima parte del processo, è possibile scegliere quali applicazioni montare. A tal proposito, in Rete è disponibile un gran numero di componenti aggiuntivi gratuiti che permettono di implementare le funzioni di ClearOS senza alcuna difficoltà.





NethServer ★★★★★

NethServer consente di impostare il proprio indirizzo IP in fase d'installazione, così da caricare il server direttamente nel browser dopo l'avvio. L'interfaccia di gestione è pulita e ben strutturata. Al suo interno sono immediatamente riconoscibili tutti gli strumenti di cui si ha bisogno. Ogni pagina contiene un pulsante che consente di accedere al supporto, così da avere sempre una spiegazione delle funzioni su cui andrete ad agire. Le pagine comprendono la gestione degli utenti, la configurazione del servizio, l'installazione del software, nonché la configurazione di rete e molto altro ancora. Impostare i vari servizi è un'operazione molto semplice. Ci sono opzioni per il backup e il ripristino delle configurazioni di sistema, ma anche funzioni per i salvataggi giornalieri che possono essere eseguiti manualmente o in modalità automatica.

Lavorare con Windows

Talvolta i server devono avere a che fare anche con Microsoft

Per quanto il nostro amore per Linux sia infinito, non bisogna dimenticare che c'è sempre un vasto numero di persone che utilizza Windows. Ciò significa che il nostro server deve essere in grado di funzionare anche con il sistema operativo Microsoft. Per quanto riguarda Web e posta, non ci sono problemi. I protocolli sono standard e il server di solito non si preoccupa della piattaforma utilizzata per comunicare con esso. Tuttavia ci sono alcuni protocolli progettati solo per Windows. Un esempio riguarda i servizi di Microsoft Exchange che vengono considerati da molti come un requisito essenziale in una rete dove sono presenti computer Windows. ClearOS dispone di **Zarafa**, un plug-in a pagamento capace di fornire **Microsoft Exchange**. Pur non essendo pienamente compatibile con tutto, funziona con la maggior parte delle piattaforme desktop e mobile. Se invece volete avere Exchange come strumento nativo, allora consigliamo di provare **OpenChange** che trovate incluso in Zentyal. NetshServer, invece, sfrutta **SOGo**, un componente aggiuntivo simile a Zarafa che però non esclude la possibilità di utilizzare anche OpenChange. Ubuntu Server, non limitandosi al contenuto di un'interfaccia Web, consente di installare qualsiasi plug-in, tra cui quelli sopra elencati.

Verdetto

ClearOS
★★★★☆
NethServer
★★★★★
TheSSS
★★☆☆☆
Ubuntu Server
★★★★★
Zentyal
★★★★★

» NethServer ha OpenChange, ma Zentyal vince per semplicità di configurazione.

TheSSS ★★★☆☆

Nonostante le sue dimensioni ridotte, TheSSS include un'interfaccia di amministrazione Web. È piuttosto semplice e consente di gestire velocemente i file di configurazione. Per eseguire alcune operazioni è necessario usare SSH. Tuttavia potrete trovare alcuni script che vi aiutano nell'amministrazione della piattaforma. Lanciate **helpme** in un terminale, così da visualizzare l'elenco dei comandi. Per una lista delle istruzioni relative al server scrivete **server**, in modo da avere a disposizione tutti i vari sotto-comandi del caso. TheSSS viene eseguito direttamente da CD o USB. Se però non avete familiarità con la riga di comando, è meglio cercare altro.



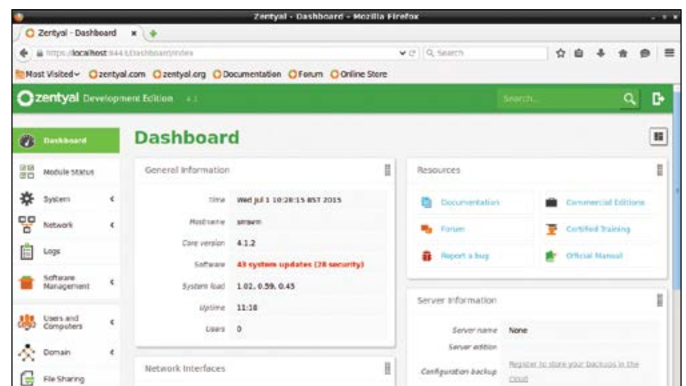


Ubuntu Server ★☆☆☆☆

L'unico strumento d'installazione remota presente in Ubuntu Server è OpenSHH. È possibile aggiungere pacchetti Zentyal, ma se avete intenzione di procedere su questa strada è meglio montare fin da subito la distro in questione. L'unico altro modo per gestire Ubuntu Server è utilizzare i Puppet. Questi, tuttavia, sono più indicati all'uso su diversi sistemi in contemporanea. Ubuntu Server è pensata per la fascia professionale del mercato e ha quindi bisogno di un SysAdmind esperto per essere gestita a dovere. Per facilitare le cose si potrebbe sfruttare **Webmin**, un tool per l'amministrazione di sistema Web-based, ma in ogni caso questa distro non è adatta a un ambiente domestico.

Zentyal ★★★★☆

Zentyal lavora su un desktop X completo, lanciando LXDE con Firefox nella pagina di login per la configurazione. È possibile utilizzare tale interfaccia da un altro computer in rete, ma niente vieta di avere accesso fisico al server. L'interfaccia non è molto intuitiva. Infatti, quando si vuole modificare un modulo, oltre a confermare le scelte premendo il tasto **Modifica**, è necessario ribadire il concetto facendo click su **Salva**. In caso contrario, qualsiasi variazione andrà persa. Una volta abituati a questo modo di fare, l'ambiente grafico diventa molto più familiare e se ne potrà apprezzare la struttura con cui è progettato. Visto che in linea c'è veramente poca documentazione in merito, riuscire ad approcciarsi facilmente alla gestione del server è sicuramente un punto di forza.



Stabilità e sicurezza

Al di sopra di tutto ci deve essere l'affidabilità

In questo frangente ci teniamo a specificare il senso con cui usiamo il termine "stabilità". Lo facciamo tenendoci stretto il concetto sviluppato da Debian, vale a dire la possibilità di mantenere lo stato delle cose nel modo più funzionale possibile e senza particolari cambiamenti. In altre parole, ogni distro server non deve sotterrarci sotto una moltitudine di aggiornamenti e correzioni di bug, ma dimostrare la sua solidità fin da subito.

Un server non ha bisogno di evolversi rapidamente, ma solo di funzionare bene e senza intoppi. TheSSS è la versione server di 4MLinux, una distro completamente indipendente. Alle spalle non ha una grande infrastruttura e quindi non è adatta a situazioni in cui la stabilità è la massima aspirazione. Si tratta di un prodotto pensato soprattutto per la messa a punto di una piccola rete domestica, magari gestita da hardware abbastanza vecchio.

Le altre piattaforme sono invece basate direttamente o indirettamente su distribuzioni principali. ClearOS e NethServer fanno capo a CentOS, vale a dire la variante libera di Red Hat Linux Enterprise. Il supporto e i relativi pacchetti per la sicurezza non mancano e sono ormai rodati. Lo stesso vale per Ubuntu Server e Zentyal che si basano sempre su Ubuntu. Questi ultimi fanno riferimento alla versione LTS che rilascia supporti validi per cinque anni.

Verdetto

ClearOS ★★★★★

NethServer ★★★★★

TheSSS ★★☆☆☆

Ubuntu Server ★★★★☆

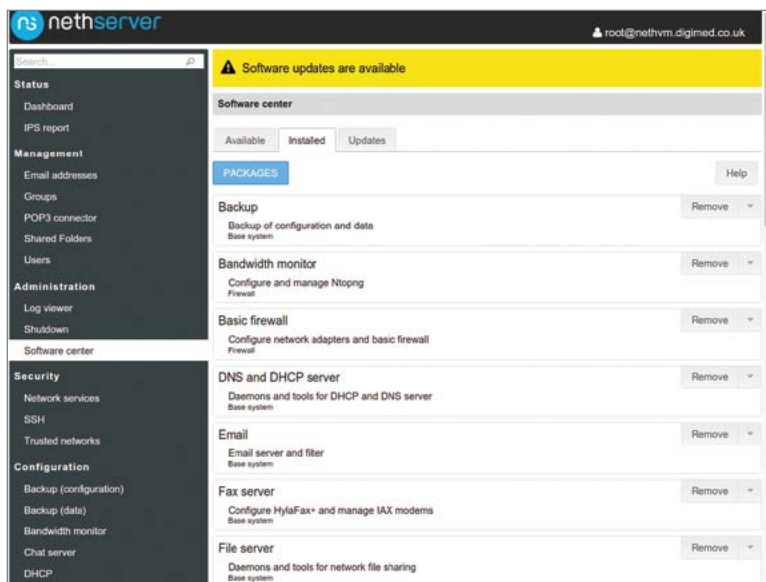
Zentyal ★★★★☆

» ClearOS e NethServer sono legati dall'accesso al software RHEL.

Documentazione e supporto

Una buona documentazione è di vitale importanza

Seppure la maggior parte degli strumenti sono gestibili da pannelli di controllo grafici, è fondamentale disporre di una documentazione appropriata per controllare i server al meglio. NethServer fornisce una discreta linea guida per l'uso dell'interfaccia. Se poi avete bisogno di maggiori info, è sufficiente consultare la documentazione completa presente nel sito Web ufficiale. Qui troverete anche una nutrita schiera di spiegazioni sulle modalità d'installazione dei software di terze parti. La guida online di ClearOS non è altrettanto dettagliata, ma è comunque molto valida per la ricchezza di materiale incluso: manuali, guide e tutorial. ClearOS è disponibile in due versioni: gratuita e professionale (a pagamento, ma comunque con possibilità di prova gratuita per 30 giorni). Il mercato di applicazioni per questa distro comprende anche software commerciali. NethServer offre supporto gratuito che però non è paragonabile a quello a pagamento. Ubuntu Server, non disponendo di alcune interfacce Web, fornisce solo di documentazione online. Non si tratta per forza di un difetto, considerando soprattutto gli alti standard a cui Ubuntu ci ha abituato nel supporto. Infatti, il materiale che viene messo a disposizione è allo stesso tempo completo e comprensibile. Trovate anche un pacchetto di supporto per Ubuntu Zentyal che può contare su una propria wiki. Questa include una sezione disponibile in svariate lingue, con documentazione ufficiale realizzata dal team di Zentyal. Il supporto fornito da TheSSS è minimale come la distro. Non aspettatevi quindi particolari informazioni sul funzionamento del server.



› ***NethServer* offre un'ottima documentazione e un buon supporto online**

Verdetto

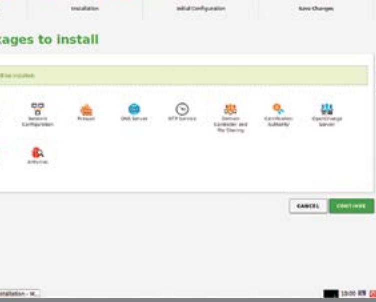
ClearOS
★★★★☆
NethServer
★★★★★
TheSSS
★★☆☆☆
Ubuntu Server
★★★★★
Zentyal
★★★★☆

» NethServer vince grazie all'ottima documentazione disponibile.

Oltre i servizi LAMP

Come sono le funzioni avanzate?

Oltre ai consueti servizi LAMP, un server può avere molteplici funzioni. Fatta eccezione per TheSSS che si ferma principalmente a pochi compiti, gli altri offrono molto di più. Tutti possono essere utilizzati come server gateway, firewall e perfino per l'uso di connessioni VPN tra gli utenti remoti e la rete. A parte TheSSS, tutte le piattaforme prese in considerazione affondano le loro radici nelle distro più famose. Zentyal non è pensato per l'uso in Internet. Infatti, non fornisce alcun supporto per il Web o servizi FTP, tra cui condivisione di file, controller di dominio, firewall, VPN e quant'altro. Bensì comprende OpenChange, un'implementazione del protocollo Microsoft Exchange che rende questo server perfetto per una rete aziendale mista. ClearOS e NethServer mettono a disposizione quasi tutte le funzioni. Entrambi sono basati su CentOS che permette loro di accedere ai tanti software disponibili per questa distro. A parte i servizi LAMP, mail e FTP, avete a disposizione un proxy server Web e mail, chat, Webmail, firewall e perfino un server per la gestione dei fax. Ubuntu Server, invece, dispone di una vasta gamma di pacchetti Ubuntu e Debian tra cui scegliere.



› ***Zentyal* dimostra che un server può essere utilizzato per molteplici scopi**

Verdetto

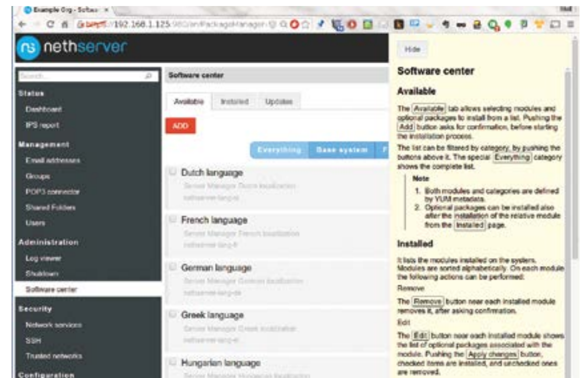
ClearOS
★★★★★
NethServer
★★★★★
TheSSS
★★★☆☆
Ubuntu Server
★★★★★
Zentyal
★★★★★

» Grazie alle sue funzioni, Zentyal va molto oltre i tradizionali servizi.

Distro Server

Il verdetto

In questo caso sarebbe troppo superficiale dire che una qualsiasi delle distro provate permette di sfruttare il miglior server. Tutto dipende dall'uso che se ne vuole fare. L'unico outsider in questa circostanza è TheSSS. Questa piattaforma non fa alcuno sforzo per somigliare a una distro completa. È piccola e leggera, ma può essere utilizzata solo per attività poco intensive e comunque non avanzate. È comunque la sola distro che può essere eseguita direttamente da CD o USB. Ubuntu Server è l'unico che non dispone di uno strumento per la configurazione Web. Se ne potete fare a meno, potrete sfruttare la potenza di questa piattaforma in modo davvero significativo. Ubuntu Server fornisce tutte le funzioni di cui si può avere bisogno. Grazie ai suoi enormi repository, così come alla semplicità nell'aggiunta dei PPA, sarete in grado di utilizzare una piattaforma potente. Il tutto con supporto LTS di 5 anni. Da non dimenticare poi la documentazione che si distingue per organizzazione e completezza. Zentyal ha due caratteristiche che lo contraddistinguono rispetto agli altri: un'eccellente integrazione in ambiente multi-OS e la mancanza di un server Web (Apache è installato solo per uso interno, ma non è disponibile tramite interfaccia). Questo indica chiaramente che il suo scopo è l'utilizzo in piccoli e medi uffici che non richiedano attività Web. Infine, visto che sostanzialmente si tratta di Ubuntu Server con front-end Zentyal, tutti i vantaggi già accomunati alla precedente distro si applicano anche in questo caso. Scegliere tra ClearOS e NethServer è molto difficile. Entrambi sono basati su CentOS 6.6 e forniscono una vasta gamma di servizi. Considerando però la parte relativa all'installazione e alla documentazione, abbiamo deciso di mettere al primo posto NethServer.



L'interfaccia di *NethServer* non è bella, ma molto funzionale

> "Se anche più di un server vi sembra adatto alle vostre esigenze, dovreste provarlo"

1° NethServer ★★★★★

Web: www.nethserver.org **Licenza:** GPL3 **Versione:** 6.6

» Anche se di poco, vince rispetto a ClearOS.

2° ClearOS ★★★★★

Web: www.clearos.com **Licenze:** Varie Open Source **Versione:** 7.1

» Basato su CentOS, vale la pena di provarlo.

3° Zentyal ★★★★☆

Web: www.zentyal.org **Licenze:** Varie Open Source **Versione:** 4.1

» Ecco come dovrebbe essere un server basato su Ubuntu.

4° Ubuntu Server ★★★★☆

Web: www.ubuntu.com/server **Licenza:** Varie **Versione:** 14.04 LTS

» È Ubuntu e funziona bene, ma non ha un'interfaccia di amministrazione.

5° TheSSS ★★★☆☆

Web: http://thesss.4mlinux.com **Licenza:** GPL3 **Versione:** 13.1

» Soluzione leggera e piacevole, ma non può essere paragonata alle altre.

A voi la parola...

Non siete d'accordo con le nostre scelte? Avreste usato altre distro? Inviate le vostre opinioni su questo confronto a: **recensioni@linuxpro.it**

Considerate anche...

Se non volete utilizzare un server pre-confezionato, potreste provare a installare l'occorrente all'interno di una qualsiasi distro. Anche quella che utilizzate per l'uso quotidiano può essere un ottimo punto di partenza. Una volta che il sistema è pronto, basterà rimuovere i pacchetti desktop uno alla volta. In alternativa, è possibile utilizzare **Gentoo** o **Arch Linux**, così da installare una distro da zero, quindi aggiungere mano a mano i componenti server necessari. Se volete sfruttare un'interfaccia di amministrazione basata su browser, **Webmin** è l'ideale. Non è particolarmente intuitiva, ma ha il pregio di essere compatibile con qualsiasi piattaforma. Scegliere una distro anziché un'altra, così come costruirsi un server da zero o sfruttare soluzioni già pronte, dipende molto dalle necessità che avete. Se siete degli smanettoni e vi divertite a realizzare un nuovo progetto, il fai da te è sicuramente da preferirsi a qualsiasi altra alternativa. LXP

Da non perdere

qBittorrent » uGet » Tmux » Qt5CT » Krita » Scribus » Kid3 » Tanks of Freedom » Dust Racing 2D » Lollypop » CVAssistant

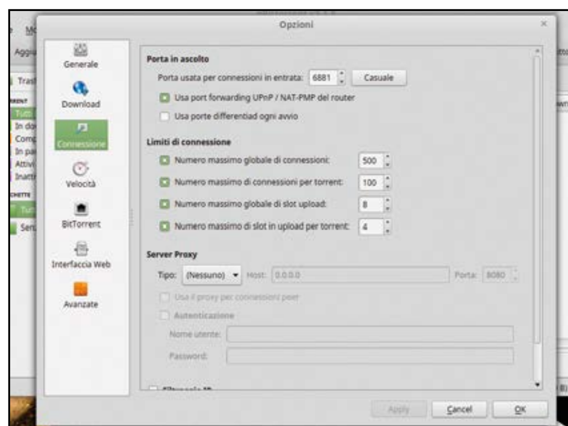
Client BitTorrent

qBittorrent

Versione: 3.2.3 **Web:** www.qbittorrent.org

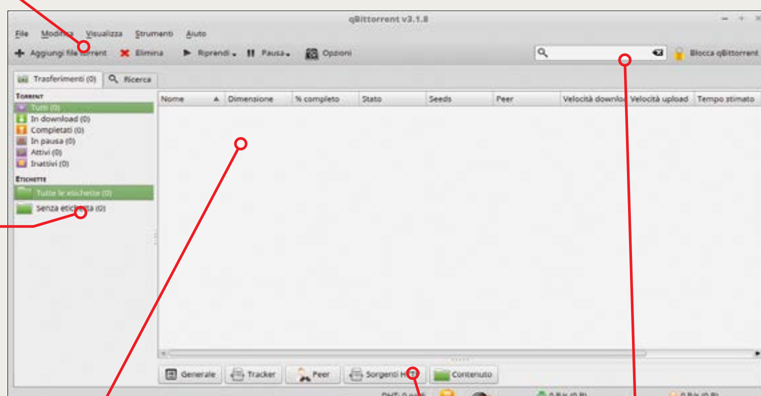
I client Open Source **qBittorrent** è disponibile praticamente per tutti i sistemi operativi. Nel caso di Linux, questo programma è quanto di più vicino e simile al software proprietario uTorrent. L'interfaccia ripercorre le stesse basi delle classiche applicazioni Torrent: i download in corso vengono riportati nella finestra principale al centro, mentre a sinistra sono presenti le varie opzioni per restringere o ampliare la visualizzazione. Il modo più semplice per usare qBittorrent è lanciare un file .torrent. Questo richiamerà il programma che inizierà così il download. In alternativa, è possibile incollare l'URL del file in questione e aspettare che il software si colleghi. qBittorrent mette a disposizione molte funzioni dedicate anche agli utenti più esperti. In primo luogo, supporta tutte le estensioni di BitTorrent, tra cui DHT, link Magnet/BitComet, indirizzi IPv6, uPnP e NAT-PMP forwarding, filtraggio IP, interfacce Ajax e molto altro ancora. L'opzione per la ricerca dei torrent è molto utile. Permette di lanciare una query su una vasta schiera di tracker. È poi possibile creare dei propri file torrent, sfruttando la procedura guidata accessibile dal menu **Strumenti**. La nuova versione 3.3 apporta diverse correzioni e aggiunge svariati miglioramenti all'interfaccia grafica che adesso può essere compilata in Qt5. I file scaricati sono evidenziati in Dolphin, Konqueror e Nautilus. In più, è adesso disponibile il nuovo stato di **Finito** per i file scaricati. È stato poi aggiunto un nuovo file **.service** per Systemd (per utilizzare qBittorrent in modalità noX) che supporta il download di documenti su determinati percorsi di rete. L'applicazione, grazie all'interfaccia Qt4/Qt5, si adatta meravigliosamente a qualsiasi ambiente desktop. Il download, oltre che in modalità simultanea, può essere configurato con funzione continua. Un aspetto, questo, di grande utilità quando avete a disposizione una connessione lenta. Per procurarsi qBittorrent basta dare un'occhiata ai repo della vostra distro e trovare il pacchetto giusto. Sono disponibili sia per Ubuntu sia per Mint nel PPA dedicato (**qbittorrent-team/qbittorrent-stable**). In OpenSUSE e Arch è poi possibile ricercare Qt5 su OBS o AUR, così da eseguire un'installazione completa.

> "qBittorrent è quanto di più vicino al software proprietario uTorrent"



› *qBittorrent* offre tutto il necessario per chi ama scaricare file torrent

L'interfaccia di qBittorrent



Strumenti di base
La barra degli strumenti contiene una serie di funzioni utilizzabili sui file in download.

Sidebar
Qui è possibile accedere alle categorie e all'elenco dei tracker disponibili.

Finestra principale
Qui viene riportato l'elenco dei file in download, il loro stato e altre informazioni molto utili.

Funzioni extra
Il riquadro inferiore mette a disposizione diversi indicatori per tenere sotto controllo lo stato della connessione.

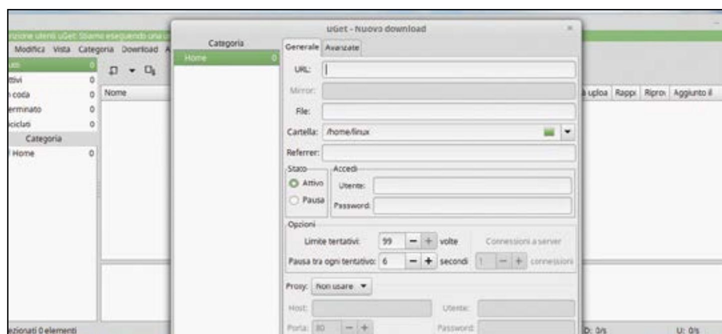
Campo ricerca
Se avete diversi torrent in download, con questa funzione potete trovare al volo quello desiderato.

Download manager

uGet

Versione: 2.0.2 **Web:** http://ugetdm.com

UGet è un'eccellente applicazione dedicata alla gestione dei download. Si tratta di un software Open Source multi-piattaforma con una vasta serie di caratteristiche. Rispetto a qBittorrent di cui abbiamo parlato nella pagina precedente, uGet è un programma più tecnico. Si rifà a un concetto d'uso piuttosto vecchio, quando ancora le connessioni dati erano relativamente lente e permettevano di gestire con difficoltà più download simultanei. Oggi, seppure non si possa più parlare di collegamenti analogici, il problema della lentezza o del risparmio in termini di tempo continua a sussistere, soprattutto per chi usa la connessione mobile con un piano dati a pagamento. Se quindi vi trovate in queste condizioni e avete bisogno di scaricare una gran quantità di file, il semplice browser non è la soluzione migliore. Ecco quindi che entra in campo uGet, capace di gestire il processo di scaricamento in modo molto più flessibile e pratico. Il software è molto leggero ed è basato su GTK3. Il download può essere messo in pausa e fatto ripartire quando volete, in più niente vi impedisce di gestire la coda dei file in modo autonomo. Infine, siete in grado di organizzare i vari documenti scaricati in categorie. uGet può essere integrato in Firefox tramite **FlashGot**. Il client supporta poi una vasta serie di protocolli, tra cui HTTP, HTTPS, FTP, BitTorrent e Metalink. Se non siete fisicamente davanti al PC, potete controllare uGet da remoto tramite la riga di comando.



› ***uGet*** **permette di controllare in modo dettagliato il download di qualsiasi file**

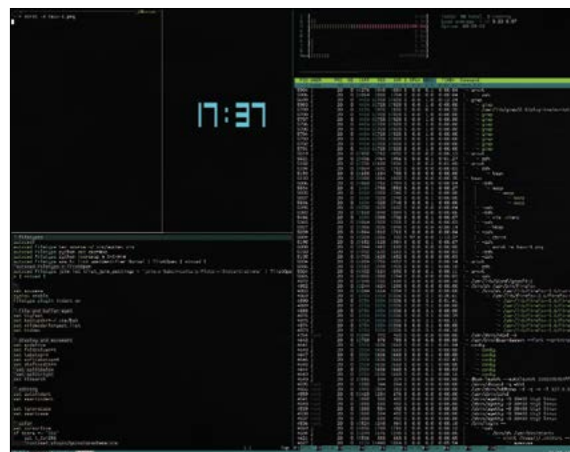
> **"uGet è un software avanzato che funziona come gestore download"**

Terminale multiplexer

Tmux

Versione: 2.0 **Web:** http://tmux.github.io

Il nuovo **Tmux** 2.0 è stato rilasciato dopo un anno di sviluppo. Si tratta di un terminale multiplexer sviluppato dalla comunità OpenBSD. Questo programma consente di dividere il proprio terminale in diversi settori, così da creare un ambiente multi-tasking ideale per gli utilizzi più avanzati. I terminali possono essere creati e controllati su una singola finestra, ma niente impedisce di sganciare ogni istanza e disporla dove si preferisce. Quando si avvia Tmux con una nuova sessione a singola interfaccia, nella parte inferiore vengono riportate alcune informazioni, tra cui le modalità di inserimento dei comandi interattivi. Il programma può essere gestito tramite una serie di scorciatoie da tastiera. Il prefisso predefinito è la combinazione **Ctrl**+**B**, a cui deve poi seguire un tasto di comando. Con **%**, per esempio, potete dividere il riquadro in due parti. Per conoscere l'elenco completo delle funzioni gestibili da tastiera, date un'occhiata al manuale disponibile con **man mux**. Per meglio comprendere il funzionamento di questa applicazione, è importante chiarire il significato di sessione, vale a dire un insieme di pseudo terminali gestiti da Tmux. Ogni sessione dispone di una o più finestre collegate. Una finestra occupa l'intero schermo e può essere suddivisa in riquadri rettangolari, ciascuno dei quali viene definito come pseudo-terminale separato (la pagina 4 del manuale spiega i dettagli tecnici di questo elemento). A una singola sessione possono poi essere collegate un numero indefinito di istanze Tmux che a loro volta sono in grado di generare una quantità illimitata di finestre. Nella versione 2.0 è stato migliorato il supporto UTF-8 e per la gestione di alcuni comandi lato server. Tmux è disponibile per svariate distro Linux. Potrete quindi installarlo in modo semplice e veloce dal vostro gestore pacchetti.



› ***Tmux*** **è un eccellente terminale multiplexer**

> **"Tmux è in grado di dividere il terminale in svariate istanze"**

Strumento di configurazione

Qt5CT

Versione: 0.19 **Web:** http://qt5ct.sourceforge.net

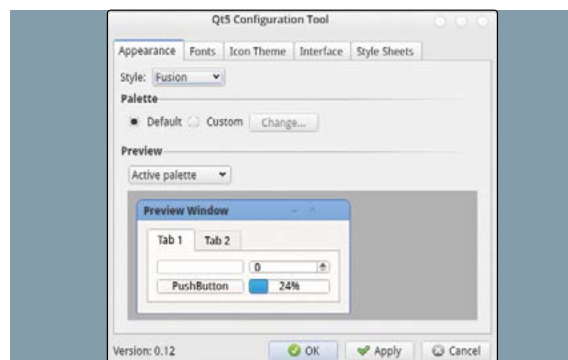
Nel momento in cui una GUI si aggiorna a una nuova versione, nella maggior parte dei casi vi troverete di fronte a un disordine primordiale. Basta ricordare le transizioni di Gnome tra GTK2 e GTK3 e quella di KDE tra la serie 3.x e la 4, per avere un'idea di quello che può succedere in questi casi.
Con il passaggio da KDE 4 a Plasma 5, siamo nuovamente nelle stesse condizioni già viste tra il 2008 e il 2009, con un'eccezione: molte più applicazioni basate su Qt stanno migrando verso Qt5. Per evitare problemi, il team di Qt ha creato **Qt5CT**, uno strumento che permette di configurare le impostazioni di Qt5 (temi, font, icone, ecc.) attraverso un Desktop Manager senza integrazione Qt. Qt5CT è utile per la maggior parte degli ambienti, tra cui Gnome, Unity, Cinnamon e KDE 4. L'applicazione dispone di sei schede per modificare l'aspetto degli widget, gestire le impostazioni del tema, gli effetti grafici, le icone e molto altro. Per impostazione predefinita, i software Qt5 utilizzano lo stile Fusion e i font tipici di KDE 4. Tutti questi aspetti possono però essere modificati proprio grazie a Qt5CT. In più, è importante ricordare che Qt5 supporta anche i fogli di stile personalizzati. L'esempio che proponiamo di seguito indica a **QLineEdits** di utilizzare il giallo come colore di sfondo e a **QCheckBoxes** di usare il rosso per il testo:

```
QLineEdit { background: yellow }
QCheckBox { color: red }
```

Il supporto per gli elementi di stile è pressoché completo, così da poter modificare qualsiasi aspetto.
Basta solo prendere confidenza con la sintassi corretta (**http://doc.qt.io/qt-5/stylesheet-syntax.html**).
Su alcuni sistemi, comunque, Qt5CT non funziona. Per risolvere il problema è sufficiente inserire le seguenti istruzioni nel file **~/.bashrc**:

```
QT_STYLE_OVERRIDE=gtkexport
QT_QPA_PLATFORMTHEME=qt5ct
```

> "Qt5CT è uno strumento avanzato, creato dal team di sviluppatori di Qt"



› ***Qt5CT* ha un aspetto professionale e fornisce una ricca serie di funzioni**

Software grafico

Krita

Versione: 2.9.4 **Web:** https://krita.org

Krita è stato per anni all'ombra della suite per ufficio Calligra (ex KOffice). Quest'ultima, pur essendo famosa per la gestione dei fogli di testo e di calcolo, ha in Krita il suo componente più importante e potente. Grazie alle sue funzioni avanzate, Krita è ben presto diventato un punto di riferimento per gli artisti digitali che usano la computer grafica per disegnare.
Un errore molto comune consiste nel paragonare Krita a Gimp o a Photoshop. Tuttavia, in questo caso, siamo su piani completamente diversi. Il software di Calligra è infatti specializzato nella pittura digitale e fornisce una serie di strumenti da usare per il disegno artistico. Gimp e Photoshop, invece, sono applicazioni per il fotoritocco e quindi molto più utilizzate dai fotografi.
Negli ultimi anni, lo sviluppo di Krita ha goduto di una notevole accelerazione che ha portato all'eliminazione e alla correzione di molti bug. In più, il supporto per il 2D è notevolmente migliorato, ponendo rimedio ai frequenti crash che spesso generava. Una delle caratteristiche più interessanti della versione 2.9.4 è il supporto per gli stili di livelli (opzione simile a quella di Photosho). Krita può quindi aggiungere ombre esterne, bagliori, smussi e molti altri effetti. Le ulteriori modifiche apportate a questa versione hanno permesso di migliorare notevolmente le prestazioni generali del programma. Adesso, infatti, è possibile eseguire anche le operazioni più esose di risorse in modo fluido e rapido.
Certo, c'è ancora molto da fare soprattutto sul fronte della compatibilità. Tutte le modifiche generate con Krita possono essere salvate solo in un formato che non viene letto da Photoshop. In più, sono ancora presenti alcuni bug che colpiscono soprattutto le maschere e i livelli. Procurarsi Krita è piuttosto semplice. Per Ubuntu e tutte le sue derivate c'è il PPA dedicato (**ppa:dimula73/krita**), mentre per il resto delle distro è possibile trovare Krita all'interno di Calligra.

> "Nella nuova versione le prestazioni sono molto migliorate"



› ***Krita*, per impostazione predefinita, utilizza un tema scuro che aiuta a concentrarsi sul contenuto**

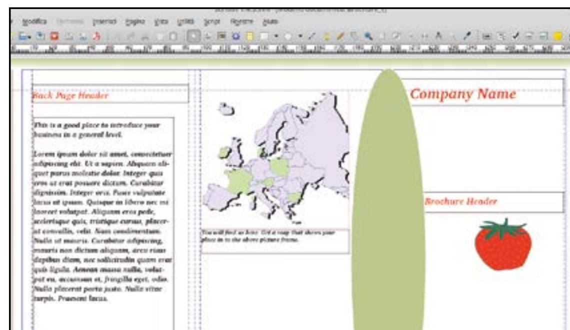
Desktop Publishing

Scribus

Versione: 1.5.0 **Web:** http://www.scribus.net

Il Desktop Publishing è sempre stato un settore in cui Linux ha mostrato la sua potenza e versatilità. Ciò nonostante, ci sono ancora poche applicazioni che possono essere utilizzate a livello professionale. Le alternative, però, non mancano e tra queste troviamo proprio **Scribus**. Progettato agli inizi del 2000, questo software è il risultato di parecchi anni di lavoro. La versione 1.5, pur avendo risolto migliaia di bug, non ha ancora ricevuto il completo benestare degli sviluppatori che non la considerano sufficientemente stabile per l'uso professionale. Tuttavia può essere utilizzata senza problemi in ambito domestico o per progetti scolastici. Tra i più importanti cambiamenti che si possono osservare nella release 1.5, troviamo il passaggio a Qt5 e l'integrazione con **UniConvertor** (consigliamo di installarlo prima di Scribus). Avete così un più ampio ventaglio di opzioni per gestire l'importazione in formati vettoriali. L'interfaccia utente è stata rivista e ridisegnata, ottimizzandone gli spazi e i vari menu disponibili. Scribus 1.5 permette poi di includere immagini bitmap all'interno del formato SLA, spostare più oggetti raggruppandoli e clonare le proprietà di un elemento master per reindirizzarle ai singoli elementi figli.

> **"Un'importante modifica tecnica della nuova versione è l'uso di Qt5"**



› ***Scribus* si sta finalmente trasformando in un software per il Desktop Publishing professionale**

Per lo più, si tratta di funzioni comuni a molti programmi professionali che avvicinano ancora di più Scribus al mondo del Desktop Publishing avanzato. L'applicazione è disponibile in quasi tutte le distribuzioni Linux, anche se non tutte offrono la versione più recente. Se non riuscite a trovare il pacchetto pre-compilato, basta seguire le istruzioni presenti nella wiki del progetto e montare il software da sorgente. Scribus ha un'eccellente documentazione e un ottimo manuale che vi aiuterà nella compilazione manuale.

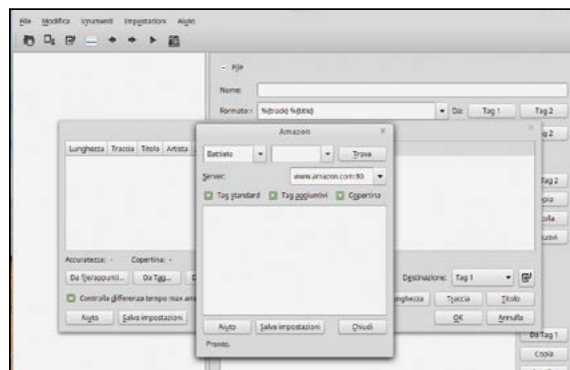
Gestore metadati per file audio

Kid3

Versione: 3.3.0 **Web:** http://kid3.sourceforge.net

Nell'era dello streaming, avere una vasta raccolta musicale memorizzata nel disco fisso può sembrare anacronistico. Tuttavia è pur sempre l'unico metodo per rimanere in possesso della nostra musica. Mano a mano che archiviate più file, diventa sempre più difficile gestirli e organizzarli in modo appropriato. Anche se siete maniaci dell'ordine e avete trovato un modo per sistemare tutte le canzoni all'interno di una cartella, ci sono sempre i metadati a creare qualche problema. Se importate i brani in un lettore multimediale, questo leggerà le informazioni contenute nel file, scartando a priori l'eventuale nome che gli avete fornito. Ecco che, nella maggior parte dei casi, l'ordine ottenuto nella cartella finisce con il diventare un caos primordiale nel momento in cui usate uno dei tanti player in circolazione. **Kid3** entra quindi in campo per fornirvi un valido aiuto nella gestione dei metadati. L'applicazione è strutturata in finestre. A sinistra trovate i file audio caricati e un mini file manager. A destra, invece, è presente un editor avanzato per la gestione dei tag. Kid3 supporta una vasta quantità di formati multimediali. Oltre al classico MP3, potete importare file Ogg/Vorbis, Opus, DSF, FLAC, MPC, MP4/AAC, MP2, Opus, Speex, TrueAudio, WavPack e WMA. Utilizzando i caratteri jolly nel campo formato, potrete definire la struttura dei tag di origine e destinazione. In questo modo, la gestione di un intero album sarà molto più rapida. Una volta presa confidenza con gli strumenti di Kid3, sarete in grado di generare tag automaticamente, convertire maiuscole in minuscole, sostituire stringhe di testo, generare playlist e molto altro ancora. L'applicazione sfrutta poi alcune caratteristiche aggiuntive, tra cui l'importazione dei metadati da svariati database online, come freedb2.org, MusicBrainz, Discogs, Amazon e altre risorse. Kid3 è nato nel 2003 ed è disponibile per la maggior parte delle distribuzioni.

> **"Kid3 permette di gestire in modo preciso i metadati dei file musicali"**



› **Con *Kid3* potete prelevare i metadati direttamente da alcune fonti online come Amazon**

Giochi Applicazioni per il tempo libero

Gioco di strategia a turni

Tanks of Freedom

Versione: Beta3.7 **Web:** http://bit.ly/TanksOfFreedom

Gli amanti dei giochi indie con grafica basilare non potranno fare a meno di appassionarsi a **Tanks of Freedom**. Si tratta di un gioco di strategia a turni, dove dovrete spingere le vostre truppe a fronteggiare il nemico. Per l'occasione, potrete usare carri armati, elicotteri e fanteria. Ogni livello è caratterizzato da un paesaggio urbano con strade, edifici e diversi ostacoli. In questo ambiente dovrete ingaggiare battaglia con il vostro avversario. L'obiettivo è conquistare il controllo di tutti i bunker nemici. Per giocare, è possibile scegliere più strategie, partendo dalle più semplici fino ad arrivare alle più complesse. Se non avete mai giocato prima a un titolo del genere, il modo migliore per iniziare è spingersi in avanti come se foste impegnati in una partita a dama. Quando le vostre unità prendono il controllo dei vari impianti (ce ne sono diversi tipi), potrete spendere risorse per creare armi ancora più potenti. Un elicottero può abbattere un carro armato, mentre tre fanti riescono a far precipitare un elicottero. I carri armati, inoltre, possono viaggiare più velocemente se gli fate percorrere le strade asfaltate. Per vincere una partita, è importante bilanciare bene le forze a disposizione. Per esempio solo la fanteria può occupare gli edifici, ma i veicoli hanno dalla loro parte la maggior potenza di fuoco. Tanks of Freedom viene fornito come un file BIN che dovrete eseguire in questo modo:

```
$ chmod +x ToF_Beta3.071_linux64.bin
$./ToF_Beta3.071_linux64.bin
```

Il gioco richiede poche risorse. Per funzionare bastano solo 33 MB di spazio libero su disco.

> "Tanks of Freedom è un divertente gioco indie basato sulla strategia"



› **Il combattimento è una cosa seria! Affrontatelo con una precisa strategia**

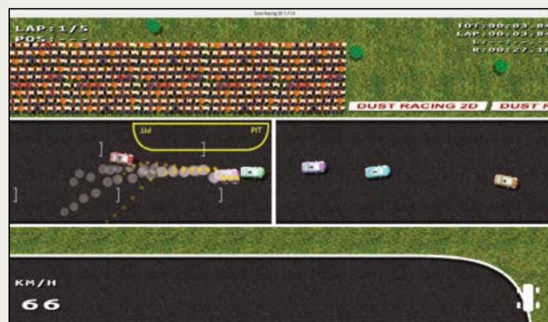
Simulatore di corse

Dust Racing 2D

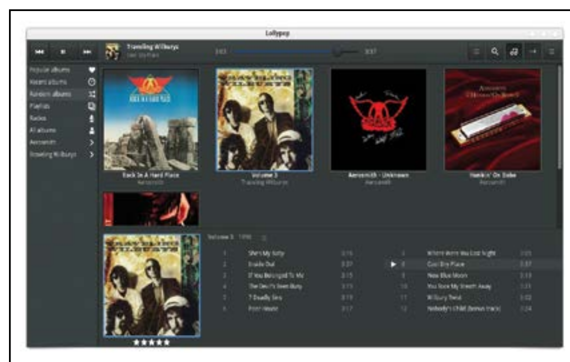
Versione: 1.11.0 **Web:** http://bit.ly/DustRacing2D

Ci sono parecchi simulatori di guida che utilizzano la visuale in prima persona. Molti meno, invece, quelli che puntano su una prospettiva diversa. **Dust Racing** fa parte della schiera di questi ultimi, poiché inquadra la vostra macchina dall'alto. Il gioco è basato su un set di livelli, di cui solo il primo è disponibile fin da subito. Per sbloccare i seguenti, bisogna arrivare tra i primi sei finalisti. La gestione dei controlli è particolarmente complessa e mantenere l'auto su strada è tutt'altro che semplice. Non supportando alcun tipo di periferica dedicata come un joypad, dovrete usare le sole frecce direzionali della tastiera. Per accelerare basta premere la freccia in alto, mentre quella in basso serve per frenare. I pulsanti destra e sinistra consentono di sterzare. La visuale sopraelevata non aiuta e in molti casi fornisce un ulteriore elemento di difficoltà. Durante le prime partite, il controllo dell'auto è davvero frustrante. Si riesce a malapena a tenerla in traiettoria sui rettilinei, figuriamoci in curva o quando si devono sorpassare gli altri concorrenti. Ci vuole tempo prima di abituarsi e non è detto che la fatica di guidare in questo modo non porti ad abbandonare il gioco dopo i primi minuti. Ciò nonostante, se avete la pazienza di imparare, potrete divertirvi non poco. Dust Racing 2D si avvia in modalità a schermo intero, ha un'ottima musica di sottofondo e livelli di difficoltà ben dosati. Potete anche giocare in due condividendo la tastiera, così da gareggiare in prove di velocità o duelli. Il gioco è disponibile per la maggior parte delle distro Linux, tra cui Ubuntu, Debian, Mageia e molte altre.

> "Se avrete la pazienza di imparare, potrete divertirvi non poco"



› **Evitare incidenti è davvero difficile**



› Semplice e inutitivo, *Lollypop* è uno dei miglior player basati su Gnome

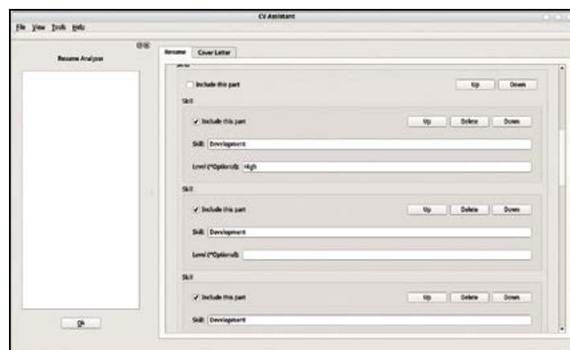
Player musicale

Lollypop

Versione: 0.9.52 **Web:** http://bit.ly/LollipopPlayer

No, non stiamo parlando del sistema operativo mobile Android 5.0. **Lollypop**, in questo caso, è un player musicale progettato su Gnome e valido per adattarsi a qualsiasi altro ambiente. Basato su GTK3, offre un'interfaccia pulita e ben strutturata che si distingue per i controlli ottimamente organizzati. Lollypop utilizza alcune parti di Totem per l'analisi dei metadati. È quindi in grado di gestire playlist e sfruttare svariate fonti multimediali. Quando avviate l'applicazione per la prima volta, la directory **home** viene aggiunta automaticamente, dopodiché il software inizia una scansione del disco fisso alla ricerca di file musicali. Le linee guida consigliano di utilizzare la funzione di ricerca automatica, ma niente impedisce di precedere anche manualmente. Nel riquadro a sinistra troverete le categorie suddivise tra album più recenti, statici, dinamici e playlist. C'è anche la categoria **Radio** che riepiloga un set predefinito di stazioni che trasmettono in Rete. Lollypop sfrutta diverse caratteristiche avanzate di GTK3, tra cui la modifica della combinazione dei colori. Premendo il pulsante **Party**, l'interfaccia cambia tema, assumendo un aspetto più scuro che permette di concentrarsi meglio sul contenuto. Un'ulteriore funzione molto utile consiste nella ricerca delle copertine dei vari album. Lollypop provvederà a fornire una serie di cover per quello specifico titolo, quindi starà a voi scegliere quella più appropriata. Nonostante sia semplice da usare e non eccessivamente carico di opzioni, Lollypop si dimostra un programma molto versatile. Il software è disponibile per svariate distro, tra cui Arch, Fedora, OpenSUSE e Ubuntu/Debian.

> **"Lollypop scansiona il disco fisso alla ricerca di file musicali"**



› Avere un CV ben fatto è fondamentale per cercare lavoro

Applicazione per la creazione di Curricula

CVAssistant

Versione: 2.0.0 **Web:** http://bit.ly/CVAssistant

Con i tempi che corrono, spesso dobbiamo essere in grado di avere più competenze professionali. Non è inusuale trovarsi a svolgere diversi lavori nei settori più disparati. La stesura di un Curriculum ad hoc per il tipo di attività per cui vi candidate è un'operazione molto importante. Se fate tutto con Word, per realizzare un CV adatto allo scopo, dovrete modificare le varie voci manualmente, eliminando o aggiungendo le competenze che vengono richieste per quello specifico lavoro. L'ideale, infatti, sarebbe avere un Curriculum sintetico, ben dettagliato e strutturato solo per l'attività per cui vi candidate. **CVAssistant** serve proprio a questo. Infatti, permette di creare in formato Word DOCX un documento con tutte le competenze ed esperienze fondate sull'annuncio a cui rispondete. L'applicazione è basata su Qt e consente di creare un Curriculum, aggiungere il proprio profilo con foto e selezionare i campi di interesse da esportare. Una caratteristica molto utile di CVAssistant consiste nell'uso della funzione **Analyzer**. Quest'ultima permette di analizzare il vostro Curriculum in base a uno specifico annuncio, così da fornirvi un giudizio di corrispondenza. A tal proposito, tenete presente che molte aziende usano un *Applicant Tracking Software* (ATS) per abbinare il proprio annuncio ai CV in arrivo, così da rilevare automaticamente le competenze d'interesse e procedere con il rispettivo colloquio. Una volta completato il CV, premete **Ctrl**+**E** per esportare il documento in formato DOCX, quindi **Ctrl**+**Maiusc**+**E** per aggiungere la lettera di accompagnamento (in formato testo). Il CV principale verrà poi salvato separatamente nel formato CVA basato su XML. Attualmente, CVAssistant non è disponibile per tutte le distro. In Ubuntu può però essere installato con l'aggiunta delle seguenti dipendenze:

```
$ sudo apt-get install qt5-qmake libqt5-dev
```

e l'uso del comando `$ qmake && make`. LXP

> **"CVAssistant è un software specializzato nella creazione di CV"**

Tutorial

I nostri esperti offrono i loro consigli di programmazione e di amministrazione del sistema

LA GUIDA DI RIFERIMENTO

Esiste sempre qualcosa di nuovo da imparare in campo informatico, soprattutto in un mondo dinamico come quello di Linux e dell'Open Source. Ogni numero di Linux Pro presenta una sezione dedicata a tutorial realizzati da esperti in moltissimi settori: programmazione, sicurezza, amministrazione di sistema, networking. Troverete informazioni utili sia che siate dei veterani di Linux sia degli utenti alle prime armi. Studieremo con cura anche le applicazioni più diffuse sia in ambito lavorativo che desktop. Il nostro scopo è quello di fornire in ogni numero il giusto mix di argomenti, ma se avete suggerimenti su temi particolari che vorreste vedere trattati, scriveteci via e-mail all'indirizzo tutorial@linuxpro.it

COME RAPPRESENTIAMO LE LINEE DI CODICE

Si presenta spesso la necessità di riportare le linee di codice di un programma. Per favorirne la lettura evidenzieremo le singole linee in questo modo:

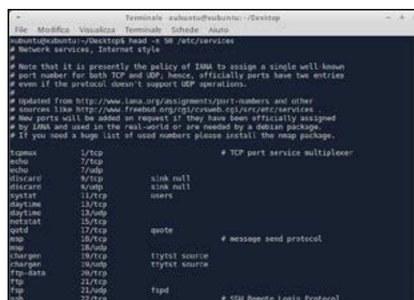
```
begin
  mniWordWrap.Checked := not
end
```

Quando una riga di codice supera la lunghezza della colonna la riporteremo su più righe utilizzando la notazione seguente:

```
printf("Vi preghiamo di inserire
una password.");
```

TUTORIAL







ACCADEMIA DEL CODICE

Sicurezza: tool e soluzioni

Rinforziamo le difese della nostra rete contro le minacce esterne con alcuni accorgimenti davvero essenziali...

Nella puntata precedente abbiamo esaminato i rischi legati all'esecuzione di servizi di rete in uno o più computer connessi a Internet senza la necessaria cornice di sicurezza. Partendo dallo schema in **Fig 1**, abbiamo realizzato un ambiente di test (per la descrizione di dettaglio si rimanda al box **L'ambiente di test** nella pagina accanto) basato su macchine virtuali Xubuntu e Damn Small Linux (DSL), eseguite in modalità live dall'hypervisor VirtualBox. La successiva configurazione di tale ambiente, riassunta nel box omonimo, ci ha consentito di simulare l'esposizione a Internet di tutti i dispositivi (le macchine **Router**, **ClientA**, **ClientB** dello schema in **Fig 1**) appartenenti a una semplice rete SOHO (acronimo di *Small Office/ Home Office*). Per aumentare il valore formativo del nostro "test bed", abbiamo provveduto ad avviare sui due client alcuni servizi di rete (come da istruzioni riepilogate nel box **I servizi dell'ambiente di test** a pagina 67):

» un server Web sulla porta 8080 di **ClientA**, simulato mediante **netcat** (il cui eseguibile è denominato, semplicemente, nc);

» un server Web, un server FTP e un demone SSH su **ClientB**, per i quali ci siamo avvalsi dei demoni messi a disposizione, anche in modalità live, dalla distribuzione DSL.

Conseguito anche questo obiettivo, abbiamo verificato come i suddetti servizi risultassero fruibili anche da parte della macchina denominata **Attacker** nello schema in **Fig 1** e simulata dalla macchina fisica: nel nostro ambiente di test, **Attacker** costituisce l'astrazione di un generico utente malizioso proveniente da Internet. In tale ottica, ci siamo cimentati nell'impresa di immaginare le possibili conseguenze nefaste dell'esposizione a Internet dei servizi offerti da **ClientA** e **ClientB**, rimandando lo studio delle possibili contromisure alla puntata attuale.

Nessuna contromisura, nessuna protezione

Prima di addentrarci nell'analisi di tali soluzioni, tuttavia, è opportuno toccare con mano l'attuale, disastrosa situazione del nostro ambiente di test, causata da una cattiva configurazione dei servizi e dall'assenza di idonee misure di sicurezza. I servizi in esecuzione su **ClientA** e **ClientB**, pensati nella nostra ipotetica rete per essere offerti ai soli componenti della LAN, risultano pienamente fruibili da Internet e, in particolare, dalla macchina **Attacker** (**Fig 1**). La macchina fisica, a cui è affidato il compito di simulare per l'appunto **Attacker**, è infatti in grado di:

» pingare i due client, come possiamo verificare avviando un terminale e digitando i comandi (**Fig 2**):

```
#ping -c 5 212.100.1.2
#ping -c 5 212.100.1.3
```

» interrogare il server Web in esecuzione su **ClientB** ricorrendo al browser, come mostrato in **Fig 3** a pagina 66;

» collegarsi in SSH a **ClientB**, mediante il comando (**Fig 4** a pagina 66):

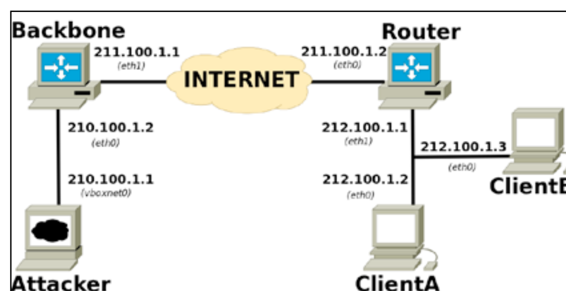
```
# ssh 212.100.1.3
```

» collegarsi in FTP a **ClientB**, mediante il comando

```
# ftp 212.100.1.3
```

(**Fig 5** a pagina 67): un limite superiore per qualsiasi altra possibile casistica. Nel nostro caso, esaminare il caso peggiore ci consente di:

» interrogare il server Web in esecuzione su **ClientA**, inserendo nella barra degli indirizzi del browser l'URL **http://212.100.1.2:8080**: in questo caso, tuttavia, il browser di **Attacker** non riceverà alcuna pagina HTML da visualizzare poiché il server Web è soltanto simulato tramite ricorso al comando **netcat** (la conferma dell'avvenuta interrogazione può essere comunque ottenuta visualizzando quanto ricevuto dall'istanza di netcat in esecuzione su **ClientA**, come mostrato in **Fig 6** a pagina 67).



› **Fig 1: Lo schema del nostro ambiente di test**

L'autore

Maurizio Russo
Laureato in Informatica presso l'Università "La Sapienza" di Roma, con una tesi sperimentale sullo stack TCP/IP del kernel Linux, è un utente del pinguino dal 2001. Nella sua carriera si è occupato di formazione, sicurezza, networking, progettazione e sviluppo di software.

```
garrick@ubuntu:~$ ping -c 5 212.100.1.2
PING 212.100.1.2 (212.100.1.2) 56(84) bytes of data.
64 bytes from 212.100.1.2: icmp_seq=1 ttl=62 time=4.20 ms
64 bytes from 212.100.1.2: icmp_seq=2 ttl=62 time=3.12 ms
64 bytes from 212.100.1.2: icmp_seq=3 ttl=62 time=2.94 ms
64 bytes from 212.100.1.2: icmp_seq=4 ttl=62 time=2.49 ms
64 bytes from 212.100.1.2: icmp_seq=5 ttl=62 time=2.92 ms

--- 212.100.1.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4003ms
rtt min/avg/max/mdev = 2.490/3.137/4.202/0.574 ms
garrick@ubuntu:~$ ping -c 5 212.100.1.3
PING 212.100.1.3 (212.100.1.3) 56(84) bytes of data.
64 bytes from 212.100.1.3: icmp_seq=1 ttl=62 time=5.00 ms
64 bytes from 212.100.1.3: icmp_seq=2 ttl=62 time=2.47 ms
64 bytes from 212.100.1.3: icmp_seq=3 ttl=62 time=2.76 ms
64 bytes from 212.100.1.3: icmp_seq=4 ttl=62 time=2.67 ms
64 bytes from 212.100.1.3: icmp_seq=5 ttl=62 time=2.74 ms

--- 212.100.1.3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 2.474/3.133/5.007/0.943 ms
garrick@ubuntu:~$
```

› Fig 2: In assenza di opportune contromisure, le macchine esterne sono in grado di raggiungere entrambi i client della rete

Ma in cosa si traduce, in termini di sicurezza, la situazione riscontrata? Essendo in grado di interagire con i servizi offerti dai due client della LAN, **Attacker** è potenzialmente in grado di sfruttare una qualsiasi vulnerabilità gravante su di essi, al fine di compromettere sia le workstation stesse, sia, eventualmente, l'intera rete locale. Si tratta di una prospettiva per nulla allettante, alla quale è necessario contrapporre adeguate misure.

Rimuovere i servizi non necessari

Se la visibilità, all'esterno della LAN, dei servizi offerti dai due client comporta un incremento del rischio di intrusioni, la soluzione più semplice e immediata non può che coincidere con la rimozione di tutti i servizi reputati non essenziali. L'idea alla base di questa strategia è banale: minore è il numero di servizi di rete in esecuzione, minore sarà il numero dei possibili vettori di un eventuale attacco. In tale ottica, l'esame dei servizi di rete in ascolto su ciascuna macchina, e la contestuale disabilitazione di quelli giudicati non fondamentali, comporta un innegabile alleggerimento degli oneri legati all'attività di messa di sicurezza della rete. Un processo di rete posto in ascolto su una porta arbitraria della nostra macchina è, infatti, potenzialmente pronto a servire le richieste provenienti da un qualsiasi indirizzo IP. In assenza di un'opportuna configurazione, il processo non è in grado di distinguere tra le richieste legittime e quelle da ritenere illegittime in quanto provenienti al di fuori della rete di riferimento. Risulta pertanto fondamentale l'attività di verifica dei servizi di rete in esecuzione su ciascuna macchina, che possiamo condurre attraverso il comando **netstat**. Si tratta di uno strumento che, grazie alla propria versatilità, può senz'altro essere annoverato tra i tool di rete più utilizzati in ambiente GNU/Linux. Attraverso **netstat**, è possibile ottenere una molteplicità di informazioni relative al sottosistema di rete di una macchina GNU/Linux: dai dati di instradamento a quelli relativi all'appartenenza a gruppi di multicast, dalle interfacce di rete disponibili alle statistiche di traffico, sino all'elenco di tutti i socket attivi. A queste funzionalità, netstat aggiunge un'invidiabile versatilità, grazie al ricorso ad appositi filtri in grado di limitare l'output (molto verboso) alle sole informazioni oggetto di interesse. Per esempio, eseguendo su **ClientA** il comando

```
# sudo netstat -natup
```

è possibile ottenere l'elenco di tutti i socket (opzione **-a**) attivi sulla workstation, relativi ai protocolli TCP (opzione **-t**) e UDP (**-u**), comprensivo di:

» indirizzi IP locali e remoti (l'opzione **-n** disabilita la risoluzione dei nomi);

» PID e nome del programma a cui fa riferimento il processo proprietario del socket. L'output del comando »

L'ambiente di test

L'ambiente di test implementa, mediante il ricorso a macchina virtuali eseguite in modalità live sull'hypervisor VirtualBox, lo schema di rete rappresentato in **Fig 1**. L'unica eccezione è costituita dalla macchina **Attacker**, simulata mediante il ricorso alla macchina fisica, al fine di contenere la quantità di memoria necessaria per le VM (appena 1.152 MB, equivalenti a poco più di 1 GB). Di seguito sono riassunte le impostazioni di creazione di ciascuna macchina virtuale

VM Router

» nome macchina virtuale: **Router**;

» sistema operativo: Linux (Ubuntu a 64 bit);

» RAM: 512 MB;

» disco fisso: nessuno;

» archiviazione: aggiungere, a creazione avvenuta, un nuovo Controller IDE alla macchina virtuale, inserendovi l'ISO di Xubuntu (da scaricare all'URL **http://xubuntu.org/getxubuntu**);

» Rete: due schede di rete, la prima connessa alla rete interna di nome **intnet1**, la seconda alla rete interna di nome **intnet**.

VM ClientA

» nome macchina virtuale: **ClientA**;

» sistema operativo: Linux (Ubuntu a 64 bit);

» RAM: 512 MB;

» disco fisso: nessuno;

» archiviazione: selezionare l'ISO di Xubuntu, secondo la medesima procedura adottata per la macchina **Router**;

» Rete: una scheda di rete, connessa alla rete interna di nome **intnet**.

VM ClientB

» nome macchina virtuale: **ClientB**;
sistema operativo: Linux (Linux 2.4 a 32 bit);

» RAM: 64 MB;

» disco fisso: nessuno;

» archiviazione: selezionare l'ISO di Damn Small Linux (da scaricare all'URL **http://www.damnsmallinux.org/download.html**), secondo la medesima procedura adottata per la macchina **Router**;

» Rete: una scheda di rete, connessa alla rete interna di nome **intnet**.

VM Backbone

» nome macchina virtuale: **Backbone**;

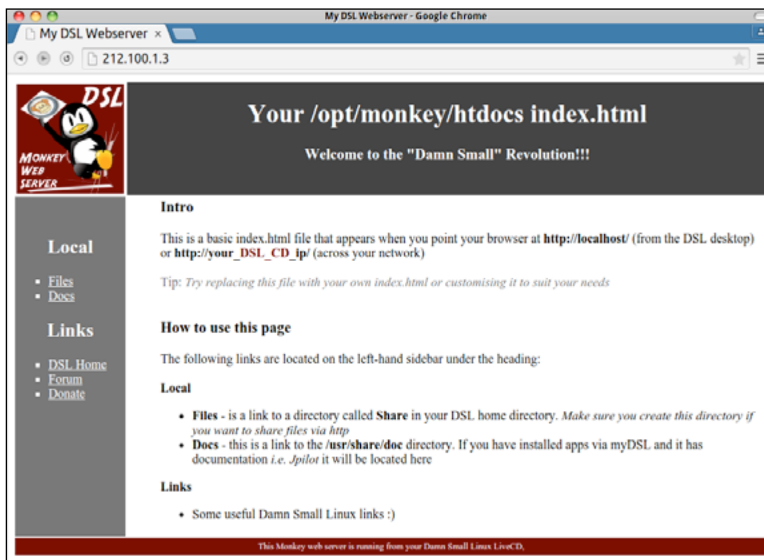
» sistema operativo: Linux (Linux 2.4 a 32 bit);

» RAM: 64 MB;

» disco fisso: nessuno;

» archiviazione: selezionare l'ISO di Damn Small Linux, secondo la medesima procedura adottata per la macchina **Router**;

» Rete: due schede di rete, la prima connessa alla rete **solo host** di nome **vboxnet0**, la seconda alla rete interna di nome **intnet1**.



› Fig 3: La configurazione iniziale consente la fruibilità, da Internet, del server Web in esecuzione su *ClientB*

(**Fig 7** a pagina 68) ci offre numerosi spunti di riflessione, a partire dall'unica connessione di cui, al momento, abbiamo la conoscenza: la connessione TCP instaurata dal browser della macchina fisica verso la porta 8080 di **ClientA**. Per inviare la richiesta HTTP riportata in **Fig 6**, infatti, il browser della nostra macchina fisica ha provveduto innanzitutto a stabilire una connessione TCP verso la suddetta porta di **ClientA**, ove abbiamo posto in ascolto netcat: non a caso, si tratta dell'unica connessione a cui netstat associa lo stato **ESTABLISHED**. Il tool ci offre tutte le informazioni necessarie a identificare univocamente la connessione:

» l'indirizzo locale (**212.100.1.2**), coincidente per l'appunto con l'indirizzo associato all'unica interfaccia di rete di **ClientA**;

» la porta locale (8080), sulla quale abbiamo posto in ascolto netcat;

» l'indirizzo remoto (**210.100.1.1**), coincidente con l'indirizzo IP associato alla macchina **Attacker**;

» il PID (2991) e il nome del programma (nc) relativo al processo che controlla la connessione.

Ancora più interessanti sono le restanti righe dell'output di netstat, in quanto fanno riferimento a socket di cui sinora eravamo del tutto ignari. In particolare, la macchina **ClientA** presenta ben quattro porte aperte, con altrettanti servizi di rete posti in ascolto di connessioni esterne:

» la porta TCP 631, aperta dal servizio cupsd;

» le porte UDP 5353, 55031, 54764 aperte dal servizio avahi-daemon;

» la porta UDP 631 aperta dal servizio *cups-browsed*.

Non tutti i servizi, tuttavia, costituiscono un potenziale pericolo. La porta TCP/631, in particolare, risulta legata all'indirizzo 127.0.0.1 ed è, pertanto, raggiungibile solo dalla macchina locale. Su tale porta, infatti, il server di stampa CUPS rende disponibile, ai soli utenti locali, la propria interfaccia Web. Si tratta di un ottimo esempio di servizio configurato con un occhio di riguardo alla sicurezza: le impostazioni di default non consentono, anche in un ambiente del tutto deficitario sotto il profilo della protezione, l'accesso diretto al servizio da parte delle macchine esterne. Possiamo verificare quanto appena asserito provvedendo ad aprire un browser sulla macchina fisica, e digitando all'interno della barra degli indirizzi l'URL **http://212.100.1.2:631**: la pagina richiesta risulterà, inevitabilmente, non raggiungibile. Al contrario, se ci spostiamo su **ClientA** e chiediamo al browser di visualizzare la medesima pagina (l'URL da digitare sulla barra degli indirizzi è, in questo caso, **http://127.0.0.1:631**), saremo ricompensati dalla visualizzazione della home page dell'interfaccia Web di CUPS (**Fig 8** a pagina 68). Se la porta TCP/631 non desta alcuna preoccupazione, le restanti, invece, meritano qualche approfondimento. L'output di netstat, infatti, evidenzia la piena accessibilità delle suddette porte anche al di fuori di **ClientA**, come testimonia il campo **Local Address**, posto genericamente a 0.0.0.0. Una valutazione complessiva, tuttavia, non può prescindere dalla conoscenza dei servizi che controllano le porte in argomento:

» **cups-browsed** è un demone utilizzato per connettere la macchina locale a stampanti remote;

» **avahi-daemon** è una soluzione a più ampio respiro, adoperabile per la connessione a risorse di rete (stampanti, ma anche file) condivise dai computer che insistono sulla medesima rete.

Qualora queste funzionalità non risultino d'interesse, è possibile provvedere a interrompere tali servizi. In ambiente Ubuntu possiamo ricorrere al comando **service** che consente di:

» conoscere lo stato (opzione **status**);

» produrre l'avvio/riavvio (opzioni **start** e **restart**);

» forzare l'arresto (opzione **stop**);

dei demoni che sono avviabili con script di init in stile System V (per intenderci, quelli posti nella directory **/etc/init.d/**) o con **job upstart** (quelli inseriti nella directory **/etc/init**). Particolarmente utile è l'opzione **–status-all** che, utilizzata da root, consente di visualizzare lo stato di tutti i servizi gestibili attraverso il comando **service** (**Fig 9** a pagina 68). Se il servizio

```
garrick@ubuntu:~$ ssh 212.100.1.3
The authenticity of host '212.100.1.3 (212.100.1.3)' can't be established.
RSA key fingerprint is 95:a3:53:ae:37:e8:0f:8c:30:f3:86:e4:8a:dc:e3:9f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '212.100.1.3' (RSA) to the list of known hosts.
garrick@212.100.1.3's password:

garrick@ubuntu:~$
```

› Fig 4: La macchina *Attacker* è in grado di connettersi liberamente al demone SSH in esecuzione su *ClientB*

```
garrick@ubuntu:~$ ftp 212.100.1.3
Connected to 212.100.1.3.
220 BetaFTPD 0.0.8pre17 ready.
Name (212.100.1.3:garrick):
```

› **Fig 5: Nella configurazione iniziale della rete anche il servizio FTP in esecuzione su *ClientB* risulta esposto a Internet**

d'interesse è incluso nella lista così prodotta, è possibile forzarne l'arresto utilizzando la sintassi:

```
# sudo service NOME_SERVIZIO stop
```

Per esempio, per arrestare il servizio **avahi-daemon**, è sufficiente digitare

```
# sudo service avahi-daemon stop
```

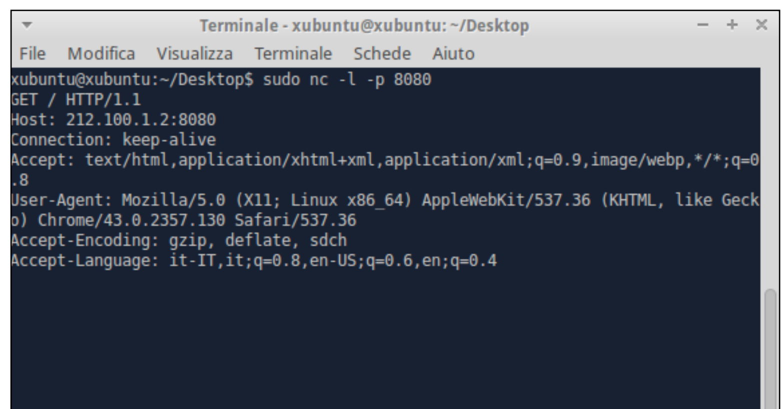
mentre per **cups-browsed**, analogamente, il comando da eseguire è:

```
# sudo service cups-browsed stop
```

In questo modo, è possibile interrompere l'esecuzione di qualsiasi servizio, ma non si impedisce allo stesso di ripartire al prossimo riavvio del sistema. Se si desidera apportare una modifica permanente alla configurazione della nostra macchina GNU/Linux, l'approccio più prudente consiste nell'effettuare una ricerca online, mirata al servizio d'interesse, nell'ambito della documentazione ufficiale della distribuzione in uso. Un utente saggio, infatti, è ben consapevole di come una modifica superficiale agli script di inizializzazione della distro possa comprometterne la stabilità o, nei casi peggiori, impedirne addirittura il caricamento.

Uncomplicated Firewall

La rimozione dei servizi non fondamentali consente una prima, significativa, scrematura delle porte di rete poste in ascolto sulla nostra Linux box. Pretendere che quest'attività comporti la rimozione di ogni potenziale rischio, tuttavia, è pura fantascienza. Nell'utilizzo comune, il ricorso ad applicativi server può scaturire da inderogabili necessità di tipo operativo: sviluppo, condivisione di file, supporto remoto, tanto per elencare tre possibili motivi per optare per i servizi in esecuzione nell'ambiente di test. Ciò non significa, tuttavia, che tali servizi debbano essere utilizzati indiscriminatamente da chiunque: »



› **Fig 6: L'istanza di netcat in ascolto sulla porta 8080 di *ClientA* riceve la richiesta inviata dal browser di *Attacker*, confermando l'esposizione del servizio all'esterno della rete LAN**

Configurazione dell'ambiente di test

Affinchè sia possibile il routing tra le macchine virtuali che compongono l'ambiente di test, basta effettuare una configurazione dei parametri di rete. L'attività di configurazione non risparmia la stessa macchina fisica, cui spetta il compito di simulare la macchina **Attacker** nello schema in **Fig 1**. Tutte le configurazioni di seguito riportate devono essere effettuate da root: in Damn Small Linux, per ottenere un terminale con i permessi di root è sufficiente cliccare su un qualsiasi punto del desktop, quindi selezionare le voci **Xshell | root access | transparent** del menu a tendina così ottenuto; in Xubuntu, possiamo anteporre al comando la stringa **sudo**.

VM Router

```
#ifconfig eth0 211.100.1.2 255.255.255.0
#ifconfig eth1 212.100.1.1 255.255.255.0
    #sysctl -w net.ipv4.ip_forward=1
#route add -net 210.100.1.0/24 gw 211.100.1.1
```

VM ClientA

```
#ifconfig eth0 212.100.1.2 255.255.255.0
#route add default gw 212.100.1.1
```

VM ClientB

```
#ifconfig eth0 up
#ifconfig eth0  212.100.1.3 255.255.255.0
# route add default gw 212.100.1.1
```

VM Backbone

```
#ifconfig eth0 up
   #ifconfig eth0 210.100.1.2 255.255.255.0
   #ifconfig eth1 up
#ifconfig eth1 211.100.1.1 255.255.255.0
   #sysctl -w net.ipv4.ip_forward=1
   #route add -net 212.100.1.0/24 gw 211.100.1.2
```

Macchina fisica

```
   #ifconfig vboxnet0 210.100.1.1 255.255.255.0
#route add default gw 210.100.1.2
```

› **Fig 7: L'output del comando *netstat* eseguito su *ClientA***

in qualsiasi scenario, compreso quello del nostro ambiente di test, è opportuno limitare al minimo indispensabile la platea dei potenziali fruitori di ciascun servizio, imponendo sulla macchina apposite politiche di filtraggio. In Ubuntu, ciò è possibile mediante il ricorso al comando **ufw**, un'interfaccia al ben più noto (e diffuso) **iptables** che si prefigge lo scopo di semplificare le attività di configurazione di un firewall *host-based* (ovvero pensato per proteggere il PC ove esso stesso è installato). All'avvio di una distribuzione della famiglia Ubuntu, ufw è inizialmente disabilitato; per ovviare a ciò è sufficiente digitare il comando:

```
# sudo ufw enable
```

Prima di effettuare questa operazione su **ClientA**, tuttavia, è opportuno simulare un secondo servizio (nella fattispecie, un server FTP), in aggiunta al server Web controllato da netcat sulla porta 8080. A tal fine, provvediamo ad aprire una seconda sessione del Terminale, digitando al suo interno il comando:

```
#sudo nc -l -p 21
```

Al pari degli altri servizi in esecuzione su **ClientA**, anche il **server FTP** è raggiungibile sia dall'interno sia dall'esterno della LAN. Possiamo verificare quanto appena asserito eseguendo sulla macchina fisica (la macchina **Attacker** nel nostro schema in **Fig 1**) il comando

```
# ftp 212.100.1.2
```

che ci ricompensa con un soddisfacente **Connected to 212.100.1.2**, a testimoniare l'avvenuta connessione alla porta 21 di **ClientA**. Attivando ufw con il citato comando **sudo ufw enable**, otteniamo invece la situazione opposta: sia il **server FTP** in ascolto sulla porta 21, sia il **server Web** in ascolto sulla 8080, risultano irraggiungibili tanto dall'esterno della rete LAN che dalle macchine locali. Non a caso, reiterando il tentativo di connessione FTP dalla macchina fisica, (ftp **212.100.1.2**) otteniamo in questo caso il messaggio (**Fig 10**)

```
ftp: connect: Connection timed out
```

dovuto per l'appunto all'intervento di ufw. Otterremmo un risultato analogo se provassimo a connetterci al server Web simulato sulla porta 8080 di **ClientA**: di base, infatti, l'attivazione di ufw scherma la macchina da qualsiasi connessione in entrata, realizzando una configurazione di rete tanto adeguata per una macchina client quanto inidonea per una workstation utilizzata per fornire servizi alla rete locale. In tale ottica, l'unica soluzione è rappresentata da una configurazione ad hoc di ufw, al fine di ottenere un filtraggio selettivo delle connessioni in ingresso a **ClientA**. Per esempio, se si desidera consentire le sole connessioni HTTP in

› **Fig 8: L'home page dell'interfaccia Web di *CUPS*, fruibile solo dagli utenti della macchina locale**

ingresso, è possibile ricorrere al comando

```
# sudo ufw allow 8080/tcp
```

che permette alla workstation di ricevere connessioni TCP (come indicato esplicitamente dal suffisso **/tcp**) sulla porta 8080, ove per l'appunto è in ascolto l'istanza di netcat che simula il Webserver. Per conseguire la raggiungibilità del server FTP, posto in ascolto sulla porta standard associata al protocollo (la porta TCP/21), possiamo invece sfruttare una sintassi più semplice:

```
# sudo ufw allow ftp
```

La stragrande maggioranza dei servizi dispone di una configurazione di default che li pone in ascolto, per ovvie ragioni di interoperabilità, sulla porta standard associata al relativo protocollo di livello application: in tale ottica, quindi, possiamo certamente affermare che la sintassi semplificata rende meno ostico l'approccio del neofita a ufw. In caso di dubbio, tuttavia, è opportuno verificare che il server d'interesse stia effettivamente attendendo connessioni sulla porta "giusta": a tal fine, è possibile ricorrere al file **/etc/services**, in cui sono definite le associazioni tra i protocolli più diffusi e le porte di riferimento. Se stampiamo a video le prime righe del file, con il comando:

```
# head -n 50 /etc/services
```

possiamo notare come (**Fig 11**) al protocollo FTP sia associata proprio la porta 21, elemento che conferma la

› **Fig 9: Utilizzato congiuntamente con l'opzione *–status-all*, il comando *service* consente di visualizzare lo stato di tutti i demoni gestibili attraverso di esso**

```
garrick@ubuntu:~$ ftp 212.100.1.2
ftp: connect: Connection timed out
ftp>
```

› **Fig 10: L'attivazione di *ufw* su *ClientA* determina l'irraggiungibilità, dall'esterno, del servizio FTP in esecuzione sul PC**

correttezza della nostra configurazione. Quest'ultima è caratterizzata, al momento, da appena due regole attive, cui si aggiunge quella di default (la cosiddetta *deny all*, ovvero nega qualsiasi connessione). Possiamo visualizzare le regole attive con il comando (**Fig 12**)

```
#sudo ufw status
```

È bene esaminare scupolosamente l'output del comando, facendo particolare attenzione all'ordine con cui sono definite le regole. Due istanze di ufw a cui sono associate le medesime regole, definite in ordine diverso, produrranno molto probabilmente un tipo di filtraggio differente: ciò è dovuto alla metodologia con cui il firewall controlla i pacchetti in ingresso. Le regole vengono applicate a ciascun pacchetto secondo l'ordine di definizione, sino a trovarne una (nel caso limite, quella di default) che corrisponda completamente ai contenuti del pacchetto esaminato. Segue che, in caso di regole configgenti, avrà la meglio quella definita per prima: è questo il motivo per cui le due regole definite manualmente sono in grado di contrastare la regola di default (eseguita da **ufw** come ultima risorsa) che invece determinerebbe l'irraggiungibilità dei servizi in esecuzione su **ClientA**. L'attività di controllo dell'ordine di definizione delle regole assume una particolare rilevanza specie negli scenari più complessi, ove sono definite restrizioni multiple, relative non solo ai protocolli esposti all'esterno, ma anche alla platea di possibili fruitori dei servizi stessi. Si pensi a due regole (che chiameremo, per semplicità, **regolaA** e **regolaB**), impostate in momenti differenti e inerenti il medesimo servizio. Se una delle due regole (che chiamiamo **regolaA**) consente la raggiungibilità indiscriminata del servizio (alla stregua della regola **ufw allow 8080/tcp** che abbiamo appena esaminato) e l'altra (**regolaB**), invece, ne vincola la fruizione all'appartenenza a una certa rete, il tipo di filtraggio ottenuto dipenderà, per forza di cose, dall'ordine di definizione! Nel caso in cui **regolaA** sia definita per prima, essa sovrasterà le restrizioni imposte da **regolaB**, consentendo l'accesso al servizio a qualsiasi indirizzo IP indipendentemente dalla rete di appartenenza. Viceversa, se invece è **regolaB** a essere definita per prima, le uniche macchine a poter accedere al servizio saranno quelle della rete specificata nella regola stessa... si tratta di una bella differenza, da cui potrebbe dipendere la sicurezza del server interessato! Questa considerazione ci offre uno spunto per risolvere le criticità di sicurezza evidenziate in precedenza nel nostro ambiente di test: possiamo infatti limitare l'accesso ai servizi offerti da **ClientA**, rendendone possibile la fruizione alle sole macchine della rete SOHO (ovvero ai soli legittimi utenti dei servizi in questione). Impostando in modo opportuno il filtro con ufw, qualsiasi PC esterno (compreso **Attacker**, che abbiamo appurato essere in grado di connettersi a **ClientA**) non potrà infatti accedere ai servizi offerti dalla workstation, determinando un incremento del livello di sicurezza globale della macchina. Tenendo bene a mente quanto appena affermato in merito all'importanza dell'ordine di definizione delle regole, dobbiamo innanzitutto procedere all'eliminazione delle regole attuali, impostate in precedenza con i comandi:

```
# sudo ufw allow 8080/tcp
# sudo ufw allow ftp
```

Queste regole consentono infatti l'accesso ai servizi da parte di qualsiasi macchina: se dimenticassimo di eliminarle, e procedessimo alla definizione di due regole più stringenti per i medesimi servizi, l'ordine di definizione comporterebbe comunque una predominanza delle regole "vecchie" rispetto a quelle "nuove", determinando l'accesso indiscriminato ai servizi da parte di qualsiasi macchina. Procediamo pertanto alla cancellazione delle regole esistenti, »

```
Terminale - xubuntu@xubuntu:~/Desktop
File  Modifica  Visualizza  Terminale  Schede  Aiuto
xubuntu@xubuntu:~/Desktop$ head -n 50 /etc/services
# Network services, Internet style
#
# Note that it is presently the policy of IANA to assign a single well-known
# port number for both TCP and UDP; hence, officially ports have two entries
# even if the protocol doesn't support UDP operations.
#
# Updated from http://www.iana.org/assignments/port-numbers and other
# sources like http://www.freebsd.org/cgi/cvsweb.cgi/src/etc/services .
# New ports will be added on request if they have been officially assigned
# by IANA and used in the real-world or are needed by a debian package.
# If you need a huge list of used numbers please install the nmap package.

tcpmux          1/tcp                           # TCP port service multiplexer
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp          quote
msp             18/tcp                          # message send protocol
msp             18/udp
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        20/tcp
ftp             21/tcp
fsp             21/udp          fspd
ssh             22/tcp                          # SSH Remote Login Protocol
```

› **Fig 11: Dalle prime righe del file */etc/services* è possibile evincere quale sia la porta standard associata al protocollo FTP**

```
Terminale - xubuntu@xubuntu: ~/Desktop
File  Modifica  Visualizza  Terminale  Schede  Aiuto
xubuntu@xubuntu:/home/xubuntu/Desktop# sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
8080/tcp                   ALLOW       Anywhere
21/tcp                     ALLOW       Anywhere
8080/tcp (v6)              ALLOW       Anywhere (v6)
21/tcp (v6)                ALLOW       Anywhere (v6)

xubuntu@xubuntu:/home/xubuntu/Desktop#
```

› **Fig 12: Il comando *sudo ufw status* ci consente di visualizzare a video le regole attualmente definite per il firewall**

mediante i comandi:

```
# sudo ufw delete allow 8080/tcp
# sudo ufw delete allow ftp
```

e verifichiamo il buon esito dell'operazione consultando l'elenco delle regole attive sul firewall, con

```
#sudo ufw status
```

Come possiamo notare, l'output del comando appare differente da quello in **Fig 12**, confermandoci l'avvenuta cancellazione delle regole. Finalmente possiamo procedere con l'impostazione di un filtro corrispondente alle "reali" necessità del nostro ambiente di test!
In un contesto come quello rappresentato in **Fig 1**, gli unici, legittimi, utenti delle applicazioni server in esecuzione su **ClientA** non possono che essere gli utenti della rete SOHO, i quali vi accedono dalle restanti macchine della LAN. Come visto in fase di configurazione, gli indirizzi IP dei componenti della rete rientrano nella subnet **212.100.1.0/24**: il filtro

```
Terminale - xubuntu@xubuntu: ~/Desktop
File  Modifica  Visualizza  Terminale  Schede  Aiuto
xubuntu@xubuntu:~/Desktop$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
21                         ALLOW       212.100.1.0/24
8080                       ALLOW       212.100.1.0/24

xubuntu@xubuntu:~/Desktop$
```

› **Fig 13: Le regole così modificate consentono la connessione a *ClientA* da parte dei soli host della rete LAN**

da impostare su ufw deve dunque consentire di stabilire una connessione TCP con **ClientA** alle sole macchine dotate di un indirizzo IP compreso in tale subnet.
Le regole da applicare sono le seguenti:

```
#sudo ufw allow from 212.100.1.0/24 to any port 8080
#sudo ufw allow from 212.100.1.0/24 to any port 21
```

La loro definizione modifica lo stato di ufw; il comando

```
#sudo ufw status
```

riflette i cambiamenti apportati, mostrando chiaramente come l'accesso ai servizi sia consentito ai soli client della rete (**Fig 13**). Verifichiamo che ciò accada davvero, provvedendo innanzitutto a controllare che l'istanza di netcat in ascolto sulla porta 21 di **ClientA** sia ancora in esecuzione: in caso negativo, occorre riavviarla con il comando

```
#sudo nc -l -p 21
```

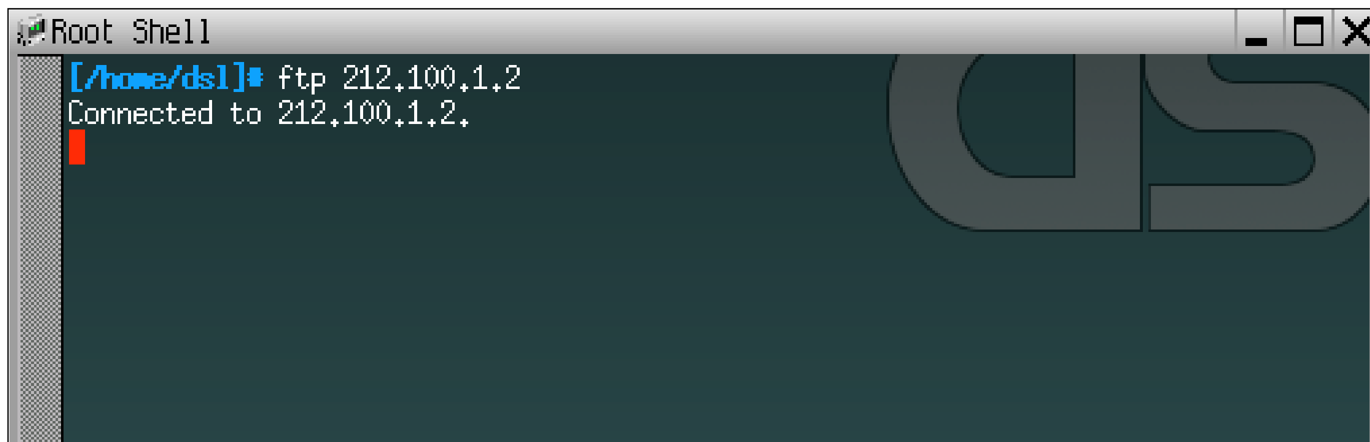
Compiuta questa necessaria operazione preliminare, possiamo dedicarci a testare attivamente la correttezza delle regole di filtraggio. Scegliamo uno dei due servizi offerti da **ClientA** (FTP, per esempio), e verifichiamo come questo sia irraggiungibile dall'esterno della rete LAN (ovvero da **Attacker**, rappresentato dalla macchina fisica) ma fruibile dai client locali (**ClientB**), eseguendo su entrambe le macchine il comando

```
# ftp 212.100.1.2
```

Il risultato è in linea con le attese: all'impossibilità di connessione, da parte di **Attacker**, corrisponde una perfetta connessione da parte di **ClientB** (**Fig 14**).
E non è tutto: il box **UFW – usi avanzati** illustra ulteriori utilizzi del firewall.

E gli altri host?

Nonostante gli innegabili miglioramenti introdotti dall'introduzione di ufw, la configurazione attuale della rete LAN rappresentata in **Fig 1** non è ancora esente da criticità. In particolare, i servizi offerti da **ClientB** risultano tuttora fruibili anche da Internet: il firewall ufw, infatti, è stato configurato sul solo **ClientA**, e pertanto



› **Fig 14: Ecco la prova dell'avvenuta connessione di *ClientB* alla porta 21 di *ClientA***

fornisce protezione esclusivamente a questa macchina. Come risolvere l'inconveniente?
La soluzione più semplice e immediata potrebbe essere quella di abilitare e configurare ufw anche su **ClientB**: sfortunatamente, però, la distribuzione Damn Small Linux non supporta questo firewall. Se anche riuscissimo a superare questa limitazione, tuttavia, l'installazione (e la configurazione) di ufw su ciascun client non costituirebbe certo la soluzione migliore al problema, né tantomeno quella più scalabile o meno onerosa sotto il profilo dell'amministrazione, specie all'aumentare del numero di workstation gestite.
L'alternativa è quella di centralizzare il filtraggio dei pacchetti, assegnando il compito alla macchina che garantisce l'interfaccia della rete con l'esterno, ovvero la macchina denominata **Router** nello schema in **Fig 1**: per scoprire come, non vi resta che attendere il prossimo numero di Linux Pro! LXP

UFW – usi avanzati

Ufw costituisce un prodotto pratico e di rapido utilizzo, in grado di migliorare il livello di sicurezza di una macchina GNU/Linux. Nonostante l'innegabile semplicità d'uso, particolarmente evidente se lo si confronta ad altre soluzioni analoghe (come per esempio **iptables**, da cui **ufw** stesso deriva), il firewall mantiene un ampio parco di opzioni, che consentono una configurazione più approfondita rispetto a quella applicata sul nostro ambiente di test. L'elenco completo delle opzioni disponibili si trova nella pagina del manuale dedicata a **ufw**, visualizzabile con il comando

```
# man ufw
```

» **--version**, in grado di restituire il numero di versione del programma (informazione particolarmente utile nel caso di eventuali vulnerabilità scoperte nel software, generalmente associate a una versione ben precisa o a un intervallo di versioni);
» **reload**, da utilizzare per riavviare ufw (operazione necessaria per rendere efficaci alcune delle modifiche alla configurazione del firewall);
» **default**, per modificare la politica di default (**allow** per consentire il passaggio, **deny** per negarlo) da applicare a una certa porzione del traffico. Ufw partiziona tutto il traffico di rete che attraversa la Linux-box in tre "direzioni", chiamate **incoming** (traffico in ingresso), **outgoing** (traffico in uscita) e **routed** (traffico instradato verso altre destinazioni): per esempio, per proteggere il computer da attacchi remoti, impedendone la raggiungibilità dall'esterno, è possibile impostare una politica di default restrittiva per i pacchetti in ingresso, mediante il comando **ufw default deny incoming**. Ogni modifica alle politiche di default, tuttavia, andrebbe attentamente ponderata, avendo cura in particolare di modificare le regole già definite, per garantirne l'adattamento al mutato scenario;
» **logging**, per abilitare o disabilitare il log dei pacchetti scartati, o per definire il livello di log. Tale livello è, di default, impostato a **low**, provocando la registrazione di tutti i pacchetti bloccati contrari alla politica di default, entro un certo limite di rate (limite imposto per minimizzare l'impatto sulle prestazioni e sulla operatività della macchina), nonché di tutti i pacchetti accettati. Sebbene questa impostazione risulti soddisfacente nella stragrande maggioranza dei casi, per motivi di debug o per approfondire il tipo di traffico ricevuto può essere utile innalzare momentaneamente il livello di log, impostandolo a **medium**. In tal caso, alle informazioni tracciate dal livello precedente si aggiungono, sempre nel rispetto del limite di rate, i pacchetti accettati sebbene contrari alla politica di default, i pacchetti invalidi e tutte le nuove connessioni. Si tratta di un potente strumento diagnostico, che tuttavia andrebbe utilizzato con cautela, in quanto è in grado di generare rapidamente log di grandi dimensioni, con il rischio di saturare lo spazio su disco disponibile. I log del firewall vengono infatti salvati nella directory **/var/log**, in file di testo il cui nome è identificato dal prefisso **ufw.log**.
I file sono organizzati per righe e ciascuna riga descrive con dovizia di particolari le caratteristiche di uno dei pacchetti oggetto di tracciatura: se da un lato ciò consente di studiare il traffico gestito dal firewall con il dovuto livello di dettaglio, dall'altro contribuisce (specie sui sistemi sottoposti a un'intensa attività di rete) a un rapido incremento dello spazio occupato su disco. Ecco per esempio una riga del file di log, registrata impostando a **low** il livello di tracciatura:

```
Aug 20 18:19:54 ubuntu kernel: [ 833.088918] [UFW BLOCK]
IN=vboxnet0 OUT=
MAC=0a:00:27:00:00:00:0a:00:27:02:09:09:09:00 SRC=212.100.1.3
DST=210.100.1.1 LEN=60
TOS=0x00 PREC=0x00 TTL=62 ID=41232 DF PROTO=TCP SPT=1030
DPT=113 WINDOW=5840
RES=0x00 SYN URGP=0
```

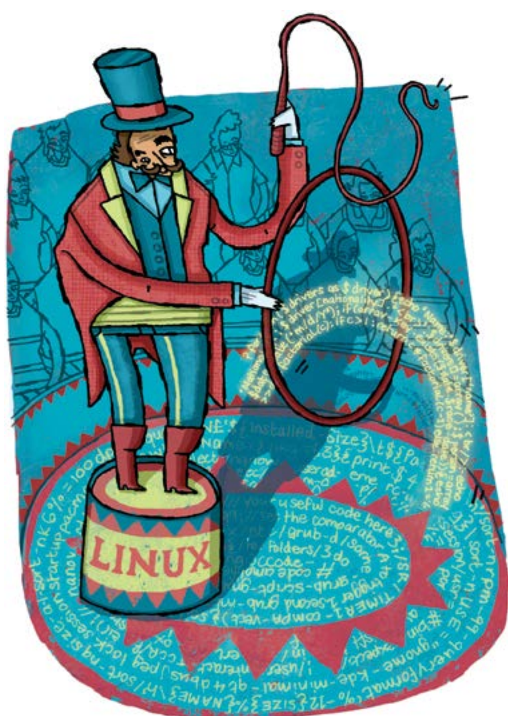
Qualora, per economizzare lo spazio su disco, si dovesse decidere di rinunciare alla tracciatura dei pacchetti (con la conseguente impossibilità di ricostruire con esattezza l'operato di ufw), è necessario impostare il livello di log a **off**, mediante il comando:

```
# ufw logging off
```

» **reset**, in grado di disabilitare il firewall e riportarlo alla configurazione di default prevista in fase di installazione: costituisce l'ancora di salvezza per i neofiti di ufw, la carta da giocare quando tutto sembra perduto!

Debian 8: un’introduzione

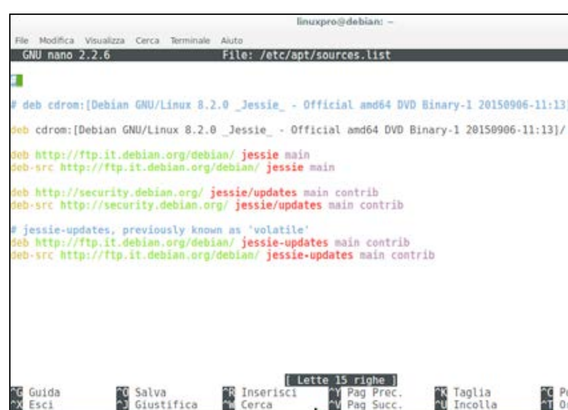
Ecco come sentirvi superiori ai vostri amici che usano Ubuntu con la vostra venerabile Debian



Tip

Se state facendo girare un paio di macchine Debian nella vostra rete casalinga, potete risparmiare un po’ di banda usando il programma **Apt-Cacher-NG** dagli archivi. Questo imposterà una macchina per essere un proxy con cache, che risulterà un repository locale.

Senza dubbio molti di voi non vedono l’ora di provare **Debian 8**, quindi quella che segue è una guida per alcuni dei task quotidiani nel più venerabile tra i Linux. Affronterete principalmente le basi ma anche alcuni trucchetti che potrebbero salvarvi la vita in alcune situazioni, in particolare nella migrazione a **Systemd**. Gli utenti che provano Debian per la prima volta sono spesso delusi che i pacchetti vengano deliberatamente mantenuti alle versioni stabili. Alcuni necessitano di pacchetti nuovi e vedrete che ci sono modi per incorporarli nella vostra installazione Debian.
Non è sempre una buona idea, tuttavia: vecchio non significa inutile, e a fronte di questo vale sempre la pena mantenere un’installazione stabile di Debian al fianco di una distribuzione più aggiornata, anche solo per avere un fallback se qualcosa dovesse andar storto. Il processo d’installazione è piuttosto lineare, posto che possiate disabilitare i meccanismi di boot sicuro. L’installer grafico è stato disegnato per essere più compatibile possibile, quindi a meno che non abbiate dell’hardware oscuro non dovreste aver bisogno di ricorrere



› **Il nostro *sources.list* standard dovrebbe assomigliare notevolmente a quello dell’immagine qui sopra**

alla modalità testuale. Se state installando un dual boot con Windows o un altro sistema operativo, l’opzione di partizionamento manuale vi permetterà di ridimensionare appropriatamente le partizioni; nel caso aveste già un layout di partizioni complicate potreste voler effettuare il partizionamento prima dell’installazione.

Il partizionamento

Non è una buona idea spostare partizioni Windows da Linux (per esempio con **Gparted**) dal momento che il bootloader di Windows è sensibile a queste cose e normalmente fallirà in seguito il boot. Evitate quindi di spostare partizioni Windows o fatelo solo dall’interno di Windows. Se state altresì ridimensionando queste cose in Gparted, assicuratevi che l’opzione **Align to** sia impostata a **None**, altrimenti la partizione sarà spostata al cilindro o MiB più vicino, causando problemi. È prudente controllare che Windows faccia ancora il boot dopo le modifiche alle partizioni, prima di esacerbare la situazione con ulteriori OS. Vale la pena notare che Debian 8.0 è anche una scelta particolarmente buona per un VPS. Ogni buon provider fornisce già immagini di Debian 8, e se non avete bisogno di grandi risorse potete ottenerne uno a poco più di 5 €/mese. Ovviamente non necessitate di un desktop su una macchina del genere, ma una volta messa in opera dal provider del VPS, potrete farvi SSH e applicare un po’ della saggezza che trovate in queste pagine. Solo perché Debian Stable snobba le ultime versioni dei pacchetti non significa che sia universalmente refrattaria agli aggiornamenti. Questi

Restate nel programma

Ci sono diversi casi in cui usare una versione più recente di un pacchetto rispetto a quella fornita da Debian risulta più gradevole, per esempio quando usate browser Web o client email che introducono caratteristiche nuove e scintillanti a ogni rilascio. I backport sono adattamenti di pacchetti da canali meno stabili. Per esempio, Mozilla mantiene un archivio di pacchetti **Iceweasel** che potete abilitare come segue. Come root, create un file ***/etc/apt/sources.list.d/mozilla.list*** contenente la riga seguente:

```
deb http://mozilla.debian.net/ jessie-backports iceweasel-release
```

Potete sostituire release con beta o aurora a seconda di quanto volete stare sul filo della novità. Questo archivio è firmato, quindi dovrete aggiungere la chiave di archivio di Mozilla al portachiavi APT. Per farlo:

```
$ wget http://mozilla.debian.net/archive.asc
$ sudo apt-key add archive.asc
```

I più paranoici tra di voi vorranno verificare la chiave scaricata. Controllate che abbia la fingerprint 06C4AE2A. Una volta aggiunta la chiave potete aggiornare l'archivio di pacchetti (vedrete alcuni riferimenti a **http://mozilla.debian.net** durante questa fase) e quindi aggiornare **Iceweasel**:

```
$ sudo apt-get update
$ sudo apt-get install iceweasel
```

Ricordate che la versione di Iceweasel integrata in Debian è parte della *Mozilla Extended Service Release*, quindi le falle di sicurezza saranno comunque corrette con la stessa urgenza; usare una versione più recente da backports non vi darà necessariamente maggior sicurezza, ma numeri di versione più grossi e più bug. C'è anche un repository ufficiale di backports per Debian con il quale potete installare pacchetti ricompilati da Testing e Unstable. Potete aggiungerlo con la seguente riga in ***/etc/apt/sources.list***:

```
deb http://http.debian.net/debian jessie-backports main
```

I pacchetti di questo repository hanno una priorità bassa (100) così da non interferire con i pacchetti da stable. Per installarli dovete fare esplicito riferimento all'archivio:

```
$ sudo apt-get install -t jessie-backports install "package"
```

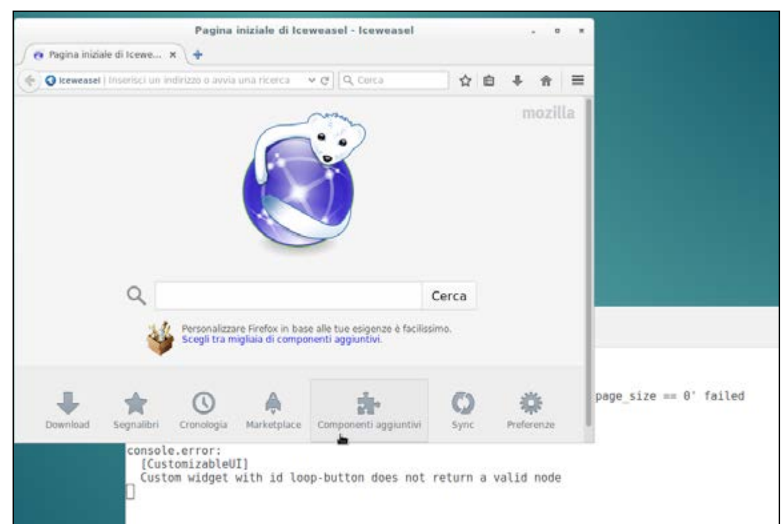
vengono applicati tanto frequentemente quanto ogni altra distribuzione, l'unica differenza è che gli aggiornamenti generalmente non coinvolgono avanzamenti di versione "major", bensì bugfix e aggiornamenti di sicurezza. Dal momento che la 8.0 è un avanzamento "major", la 8.1 conterrà diversi di questi fix. Non c'è bisogno di reinstallare o di altre cose così drastiche, poiché saranno tutti disponibili nei repository. Mantenere aggiornati i vostri pacchetti è vitale ma anche banale. Debian mantiene una lista di repository nel file ***/etc/apt/sources.list***. Potete aggiungerne altri a questo file o nella directory ***/etc/apt/sources.list.d***. Se avete installato da CD il vostro **sources.list** conterrà una riga che comincia con deb cdrom:. Questo indica al gestore di pacchetti che può recuperarli anche dal CD, risparmiando un po' di tempo dopo l'installazione, se avete una connessione lenta; una volta sincronizzati i repository la maggior parte dei pacchetti sarà da aggiornare, tuttavia, quindi potete anche commentare questa riga.

Mantenersi aggiornati

Ogni riga nel file **sources.list** specifica un repository particolare. La riga comincia con deb o deb-src che determina se l'archivio è per pacchetti binari o sorgenti. Il secondo campo è l'URL, il terzo la distro, che sarà jessie se avete appena installato. Potete anche specificare una branch qui, come stable o testing. Jessie è la release attuale, ma in un paio d'anni verrà stabilizzata Stretch (la release attualmente in testing). Sostituire il nome jessie con un nome di canale come stable in sources.list non avrà un effetto immediato, ma quando Stretch verrà stabilizzata verrà automaticamente aggiornata. Lasciare il nome così com'è indica che continuerete a seguire la release Jessie fino al termine del suo ciclo (solitamente cinque anni dopo il rilascio). Il resto della riga dell'archivio indica i componenti da usare in quel repo. Dopo l'installazione è selezionato solamente il componente main. Potete aggiungere non-free ai repo per includere vari software proprietari (come il plug-in Flash, ma non usatelo! o driver proprietari). Ci sono anche alcuni componenti appartenenti a contrib. Tali pacchetti sono aderenti alle *Debian Free Software Guidelines* (DFSG), ma dipendono da software che non lo è. Noterete che alcune righe in **sources.list** fanno riferimento a jessie-updates. Queste incorporano aggiornamenti che vale la pena di installare prima dell'aggiornamento seguente (in cui diventeranno parte integrante di Jessie). Potete anche aggiungere il nome proposed-updates per ottenere aggiornamenti programmati per il prossimo rilascio, ma tenete a mente che questi non sono stati pienamente testati. Periodicamente, e ogni qualvolta aggiornate ***/etc/apt/sources.list***, dovreste lanciare (come root) # apt-get update per aggiornare l'elenco locale dei pacchetti disponibili. Dovrete usare il comando su assieme alla password di root impostata durante l'installazione per guadagnare inizialmente accesso come root, dal momento che Debian non imposta sudo per voi. Se venite da Ubuntu, sudo è probabilmente un comando che usate spesso. Impostarlo su Debian non è difficile, lanciate il seguente come root per modificare il file con **nano** (potete usare l'editor di default, **vi**, se proprio volete, ma non è esattamente adatto ai novizi):

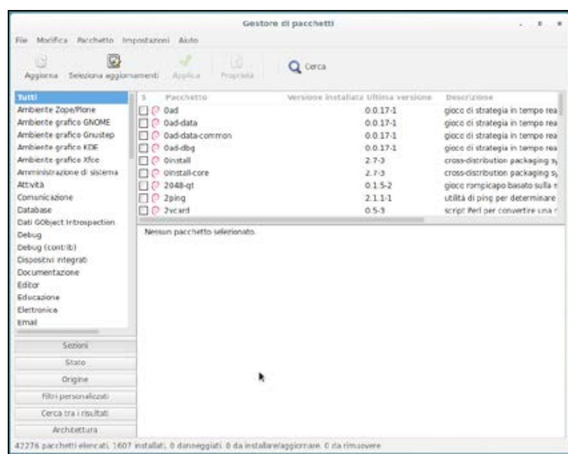
```
# EDITOR=nano visudo
```

Il programma **visudo** modifica e controlla la sintassi del file ***/etc/sudoers*** cosicché eventuali errori non risultano nella perdita dell'accesso a sudo per tutti gli utenti. Aggiungete la riga seguente, sostituendo user con il vostro username: »



› Dal momento che il logo di *Firefox* non può essere ridistribuito, Debian ha fatto un rebrand del browser come *Iceweasel*. Usando questo tutorial potete aggiornarlo alla versione 38

»
```
user    ALL=(ALL:ALL) ALL
```
Ora uscite con **Ctrl**+**c** e premete **y** per salvare. Viene salvato un file temporaneo per essere controllato da visudo e questo, se approvato, sovrascrive ***/etc/sudoers***. A questo punto non avrete mai più bisogno del comando su e potete aggiornare il vostro sistema esattamente come farebbe un utente Ubuntu:
```
$ sudo apt-get upgrade
```
Talvolta APT riporterà che alcuni pacchetti non saranno aggiornati per qualche genere di conflitto di dipendenza. Questo succederà se, per esempio, state aggiornando il canale stable in ***/etc/apt/sources.list*** nel momento in cui viene rilasciata una nuova release. La soluzione è usare il controllo di dipendenze "smart" con
```
$ sudo apt-get dist-upgrade
```
Potete usare apt-cache per cercare nella cache locale dei pacchetti disponibili. Per esempio, per cercare le descrizioni di pacchetti che corrispondono a **chromium**:
```
$ apt-cache search chromium
```
La gestione di pacchetti di Debian è capace di ben altro, oltre a questo. La suite **Apt** funge da front-end per lo strumento **dpkg**, astraendone tutta la complicata risoluzione di dipendenze e il resto. Potete controllare le informazioni di stato su un particolare pacchetto, in questo caso il browser Web **Iceweasel**, con:
```
$ dpkg -l iceweasel
```
L'ultima riga assomiglierà a qualcosa del tipo
```
ii          iceweasel 31.7.0esr-1~d          amd64     Web
browser...
```



› ***Synaptic* è pratico per controllare i pacchetti installati o cercarne di nuovi. Ma i power user non sono interessati a questo genere di interfaccia grafica**

a meno che non abbiate rimosso Iceweasel (che è parte dell'installazione di default) dal vostro sistema. Il codice a due lettere sulla sinistra indica lo stato del pacchetto: la prima i indica che è desiderata l'installazione del pacchetto, la seconda indica che è effettivamente installato. Se un pacchetto è in uno stato di inconsistenza, dpkg mostrerà un terzo carattere R indicando che è necessaria una reinstallazione. Talvolta queste riparazioni possono essere fatte automaticamente con
```
$ sudo apt-get install -f
```
che tenta di consolidare il vostro albero di pacchetti.

Rimuovere pacchetti

I pacchetti possono essere rimossi (apt-get remove) o epurati (apt-get purge): quest'ultima condizione indica che vengono rimossi anche i file di configurazione. Per i pacchetti epurati lo stato sarà visualizzato da dpkg -l come pp, mentre quelli rimossi ma con i file di configurazione ancora presenti come rc. Mano a mano che vengono installati nuovi pacchetti, alcune dipendenze potrebbero diventare orfane, quindi non più richieste. APT vi dirà di usare
```
$ sudo apt-get autoremove
```
per cancellarli, e dovreste farlo. La directory di cache dei pacchetti ***/var/cache/apt*** altresì si riempirà nel tempo. È buona regola pulirla periodicamente con
```
$ sudo apt-get autoclean
```
Oltre al comando dpkg a basso livello, c'è anche il programma di più alto livello **Aptitude**. Anche se potete usarlo al posto di apt-get per le installazioni e rimozioni quotidiane, è sconsigliabile mescolare i vari frontend, dal momento che registrano le proprie azioni in posizioni diverse. Una delle caratteristiche più belle di **Aptitude** è la capacità di vedere perché è stato installato un particolare pacchetto, per esempio se voleste vedere perché è stato installato il server **Xwayland**:
```
$ aptitude why xwayland
```
Stavate usando il desktop **Cinnamon**, che dipende dal pacchetto gnome-session-bin (Cinnamon usa ancora molte componenti di Gnome dietro le quinte) e in questo sta la dipendenza di **Wayland**. Per quel che vale, passerà un po' di tempo prima di vedere un desktop Wayland funzionante in Debian o in qualsiasi altra distribuzione. Se avete problemi nell'installazione di un pacchetto, Aptitude ha anche un comando why-not. Non c'è (grazie al cielo) un equivalente dell'**Ubuntu Software Center**, ma **Synaptic** supplirà a tutti

Systemd

A meno che non abbiate vissuto sotto un sasso per gli ultimi anni, saprete che alcune persone non sono contente che **Systemd** stia prendendo piede in praticamente ogni distribuzione Linux. Rumore a parte, ci sono alcune critiche ragionevoli a questo sistema e alcuni potrebbero volerlo sostituire. Tenete comunque a mente che anche se per ora Systemd può essere rimosso e sostituito con il vecchio **SysVinit**, non ci sono piani per mantenere il vecchio sistema più in là del rilascio di Jessie. Notate anche che Gnome (e per estensione Cinnamon) dipende da Systemd, quindi bilanciate il vostro desiderio di tali desktop con quello di eliminare Systemd. Il modo più pulito di rimuovere Systemd è usare una riga di preseed all'inizio del processo di installazione. Premete **Tab** nell'opzione di menu dell'installazione grafica e aggiungete al comando di boot:
```
preseed/late_command="in-target apt-get
install -y sysvinit-core"
```
Se avete già installato, il processo di rimozione non è così complicato. Per prima cosa installate i pacchetti di **SysV**:
```
$ sudo apt-get install sysvinit-core sysvinit
sysvinit-utils
```
Ora fate un reboot così che Systemd non sia più in memoria, quindi eliminatelo, possibilmente bruciando della salvia bianca:
```
$ sudo apt-get remove --purge --auto-remove
systemd
```
Potete usare il pinning di apt per assicurarvi che Systemd non cerchi di rientrare dalla finestra: come root modificate il file ***/etc/apt/preferences.d/systemd*** inserendovi:
```
Package: systemd
Pin: origin
Pin-Priority: -1
```

i vostri requisiti per la gestione grafica dei pacchetti. L'installazione standard di Debian è pensata per essere meno problematica possibile. Detto questo, ci sono alcune parti di cui potreste probabilmente fare a meno, a seconda di come pensate di usare la vostra macchina. Pacchetti superflui occuperanno spazio su disco e rallenteranno l'avvio, ma non siate troppo precipitosi nel cancellare cose. Per esempio, potreste decidere di non aver alcun bisogno del server mail **exim** ma alcuni demoni lo usano per inviare notifiche allo spool di mail locale, quindi potete disabilitarne solo la funzionalità con

```
$ sudo systemctl disable exim4
```

Ridurre i servizi all'avvio

Allo stesso modo potete mettere il veto ai servizi **modemmanager** e **speech-dispatcher**, andando a recuperare circa cinque secondi al boot (potete vederlo con il comando `systemd-analyze`. Debian ha adottato Systemd come sistema di init, se volete veramente disabilitarlo seguite le istruzioni nel box. Nel futuro in vista sembra che il debian Project manterrà compatibilità con il sistema **SysV init**, quindi gli script dovrebbero continuare a funzionare e Systemd manterrà i collegamenti in **/etc/rc?.d**, quindi se potete abilitare il server SSH con

```
$ sudo systemctl enable ssh
```

come per magia apparirà il link relativo, così:

```
$ ls -l /etc/rc3.d/S02ssh
lrwxrwxrwx    1    root    root ...
S02ssh->../init.d/ssh
```

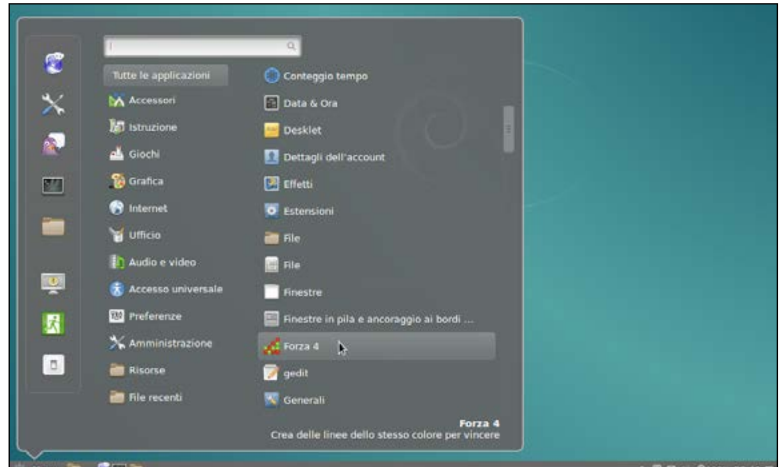
Se disabilitate tale serivizio, spariranno questo link e tutti gli altri collegati. Gli script di init di Systemd sono un po' più carini degli equivalenti SysV. Sono formattati (male) come i file INI di Windows. Troverete alcuni file di Systemd nella directory **/lib/systemd/system**. Dal momento che Systemd è piuttosto intelligente riguardo la gestione delle dipendenze, i file unità sono solitamente molto più semplici degli script SysV. Confrontate per esempio il file del servizio SSH con la sua controparte **/etc/init.d/ssh**: il file Systemd consta di 15 righe contro le 154 di SysV. Quando abilitate tali servizi con `systemctl`, vengono creati i link nella directory **/etc/systemd/system**.

Driver proprietari

Anche se siete incoraggiati a restare con i driver liberi e Open Source ovunque possibile, esistono alcune schede grafiche (in particolare le Nvidia) che si comportano molto meglio se usate con driver proprietari. Il primo passo è aggiungere i componenti `contrib` e `non-free` al repository `jessie` (la prima riga notevole in **/etc/apt/sources.list**). Quindi dovrete aggiungere la versione corretta del pacchetto **linux-headers**. Se avete fatto di recente un `apt-get upgrade` e non avete mai fatto un reboot, vale la pena di farlo così che i driver siano compilati per il kernel in esecuzione al prossimo reboot e non per quello attuale. Se non ci sono menzioni di **linux-image** o **linux-headers** nell'update, potete procedere senza reboot. Procedete e recuperate gli header del kernel corretti con

```
$ sudo apt-get install linux-headers-$(uname -r)
```

Jessie supporta due rami per il driver proprietario di Nvidia, 340 e 304 e solamente la serie 14.9 per il driver Catalyst. Se sono troppo arcaici per voi, il che è probabile, potrebbe valer la pena usare i pacchetti dal sito del produttore, ma è



› **Il desktop di *Cinnamon* è disponibile per l'installazione, è solo la versione 2.2 ma sembra molto buona!**

un modo molto non-Debian di fare le cose, quindi procedete con attenzione. Alternativamente potete esplorare l'aggiunta di pacchetti dal canale Experimental. Questo è un processo piuttosto univoco e per la maggior parte oltre l'argomento di questo tutorial. Più avanti saprete come recuperare e scegliere pacchetti da canali meno stabili (un processo chiamato *apt pinning*), ma per ora ammettiamo che siate contenti dei pacchetti forniti da Debian. Per le schede Nvidia e il driver 340 (ultima release a supportare le CPU GeForce di serie 8 e 9) installate il pacchetto come segue:

```
$ sudo apt-get install nvidia-kernel-dkms
```

Il *Dynamic Kernel Module Support* (DKMS) è un sistema utilissimo per ricorstruire automaticamente i moduli a ogni aggiornamento del kernel. Senza questo, chiunque non tenga d'occhio gli aggiornamenti dei pacchetti e reinstalli manualmente tali pacchetti troverà spesso problemi con la grafica. Se volete invece il vecchio driver 304 (che supporta praticamente le stesse schede del 340) usate **nvidia-legacy-304xx-kernel-dkms**. **X.org** non rileverà il driver nvidia, quindi dovrete creare un file **xorg.conf** minimo lanciando

```
$ sudo nano /etc/X11/xorg.conf.d/20-nvidia
```

per modificare un nuovo file, che è un modo più pulito di fare le cose. Inserite il contenuto seguente:

```
Section "Device"
        Identifier "La mia GPU"
        Driver "nvidia"
EndSection
```

Ora riavviando dovreste sentire quel brivido lungo la schiena che vi ricorda che ora il kernel è contaminato da un'essenza aliena. Notate che il driver Catalyst non è compatibile con il desktop Gnome, quindi nel caso restate al driver libero. Altrimenti, per installare il driver Catalyst usate:

```
$ sudo apt-get install fglrx-driver
```

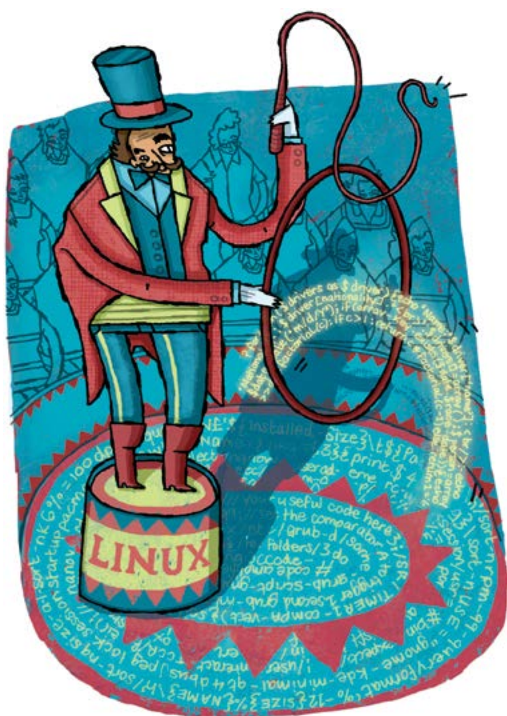
che per magia lancerà tutta la combriccola DKMS. Quindi, come root, create il file **/etc/X11/xorg.conf.d/20-fglrx.conf** con il seguente contenuto:

```
Section "Device"
        Identifier "La mia GPU"
        Driver "fglrx"
EndSection
```

E questo conclude il vostro viaggio in Debian. Assicuratevi di leggere il wiki di Debian (**https://wiki.debian.org**) per una guida su come continuare l'avventura nel mondo della spirale rossa. LXP

IPv6: come restare connessi

Se avete pensato a costruirvi un riparo di fortuna per l'imminente esaurimento di IPv4, questo tutorial vi salverà



Nel febbraio 2011 la IANA (*Internet Assigned Numbers Authority*) ha distribuito i cinque blocchi /8 rimanenti (ognuno comprendente 16 milioni di indirizzi e ottenuti raschiando il fondo del barile) tra i cinque Registri Internet Regionali (RIR). Ora non hanno più nulla da dare, fine. Stop. Sicuro, voi potrete ancora ottenere indirizzi IP per nuovi dispositivi e altro, ma le forniture dei produttori si esauriranno e non ci sono altri rifornimenti. Non fatevi prendere dal panico, comunque: diversi anni di studi e test hanno prodotto un protocollo di indirizzamento sostitutivo considerato stabile in Linux dal 2005. Alcuni grandi nomi hanno già attivato le loro reti **IPv6**. Se abitate in Italia è possibile che il vostro ISP non vi fornisca ancora la connettività IPv6 (lo fanno in pochi, e quelli tra di voi così fortunati non hanno bisogno di questo tutorial), ma potete aggirare questo ostacolo con un tunnel. È importante notare che non ci sono piani per staccare la spina a IPv4: i due protocolli possono convivere fianco a fianco.
Le schede di rete possono avere entrambi i tipi di indirizzo e i router (adatti) possono gestire entrambi i tipi di traffico. Dopo l'esaurimento, tuttavia, i nuovi host saranno accessibili solo tramite IPv6, quindi coloro con la sola connettività IPv4 non potranno più accedere a parte di internet (e viceversa da un certo punto di vista). Anche se alcuni dispositivi di rete (come i router) dovranno essere aggiornati, le schede di rete e gli switch funzioneranno perfettamente con IPv6: tutto quello che fanno è trasmettere frame Ethernet, e sono indifferenti alla natura del contenuto a più alto livello contenuto in questi. IPv4 usa 32 bit per indirizzare i nodi, con la convenzione di dividerli in quattro gruppi di otto byte (un byte o otteto) e scriverne la rappresentazione decimale separata da punti. Lo spazio di indirizzi IPv4 permette quindi circa 4,3 miliardi di indirizzi, che per coincidenza è circa uno per ogni persona sulla Terra che dispone di accesso a Internet. Mano a mano che si connettono più indirizzi a internet e il NAT, anche effettuato a livello di provider, non sembra risolvere le cose, IPv4 va verso l'esaurimento. Gli indirizzi IPv6, per confronto, usano 128 bit per ogni indirizzo, il che implica che difficilmente si esauriranno in breve tempo (ce n'è abbastanza per assegnarne uno a ogni atomo sulla faccia della Terra). La notazione standard è partizionare un indirizzo in otto gruppi di 16 bit ognuno. Ogni gruppo è scritto come quattro cifre esadecimali e separato dal successivo con due punti.

Tanti esadecimali

Dal momento che 32 cifre esadecimali e sette due punti sono difficili da scrivere, ci sono un paio di scorciatoie. La prima: ogni zero iniziale può essere omesso per ogni gruppo, quindi **0123** può essere scritto **123**. La seconda: una serie di **0** adiacenti (come **0000**) può essere rimpiazzata da **::**. Come esempio, considerate l'indirizzo di loopback (analogo di **127.0.0.1** in IPv4) che, quando abbreviato dalla prima regola, è **0:0:0:0:0:0:0:1**. Questi primi sette gruppi possono essere scartati e dovrete scrivere solamente **::1**. Notate che potete usare la notazione con il doppio due punti solo una volta per indirizzo, poiché applicazioni molteplici sarebbero ambigue. Un indirizzo IPv6 è diviso in due parti: i primi 64 bit (o quattro gruppi) formano il prefisso di rete e gli ultimi 64 sono definiti l'identificatore di host. Il prefisso dei rete è ulteriormente diviso in un prefisso di routing e un ID di sottorete, ma non preoccupatevene ora. Se state usando una distro aggiornata, probabilmente avrete già pieno supporto a IPv6. Potete controllare con il comando **ip** del

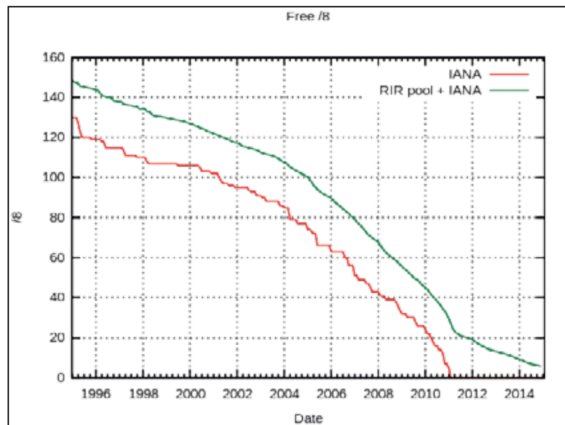
pacchetto **iproute2**:

```
$ ip a
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu
1500 qdisc pfifo_fast state UP group default qlen 1000
   link/ether 08:00:27:0a:03:45 brd ff:ff:ff:ff:ff:ff
   inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic
eth0
      valid_lft 86387sec preferred_lft 86387sec
   inet6 fe80::a00:27ff:fe0a:345/64 scope link
      valid_lft forever preferred_lft forever
```

Ecco, la scheda Ethernet (**eth0**) ha un indirizzo **inet6**. Non è tuttavia un indirizzo globalmente accessibile: il prefisso **fe80::** (dove **::** implica tre gruppi di zeri) implica che è un indirizzo locale. Qualcosa di simile alla famiglia IPv4 **169.254.x.x** per l'autoconfigurazione, che potreste aver visto quando qualcosa è andato male con il vostro server DHCP. Se confrontate l'identificatore con l'indirizzo MAC (**08:00:27:0a:03:45**) dovreste notare più di qualche analogia. Non è una coincidenza, dal momento che il link locale è generato inserendo **ff:fe** nel mezzo dell'indirizzo MAC e computando il settimo bit più significativo della stringa risultante (che aggiungerà o toglierà due dalla seconda cifra esadecimale). L'indirizzo link locale è necessario per ogni host IPv6, le applicazioni vi fanno affidamento. Oltretutto, dal momento che gli indirizzi MAC sono unici (quasi, perché possono essere costruiti o spoofati), gli indirizzi link locali sono unici per ogni dispositivo. Tale metodo di costruire l'identificativo di host, usato anche in alcuni tipi di connessioni non locali, può essere legato a dispositivi individuali e quindi a individui. Fortunatamente esistono dei meccanismi di aumento della privacy che possono generare identificativi effimeri tanto frequentemente quanto voluto.

Configurare il vostro viaggio

Naturalmente il vostro router casalingo potrebbe non essere configurato per fornire (o supportare) alcun genere di connettività IPv6. Non disperate però, perché potete impostare facilmente una rete IPv6 interna o una sola macchina (se siete pigri) o usare un tunnel per parlare in IPv6 al mondo esterno. Il vostro router permetterà il traffico IPv6 nella rete interna, anche se non potrà farne il routing all'esterno o addirittura sapere cosa sia. Per prima cosa



› **Ce n'erano molti, e ora non ne è rimasto nessuno. La stessa storia potrebbe essere vera per tutte le risorse della Terra**

dovete assegnare un prefisso ULA (*Unique Local Address*) alla macchina della rete interna che fungerà da router. Quindi potete impostarla per manifestarsi e fornire informazioni SLAAC alle altre macchine. I prefissi ULA vengono da un blocco riservato (inizia con **fd**) che non viene indirizzato verso l'esterno. Potete generare il vostro all'indirizzo **www.simpledns.com/private-IPv6.aspx** o inventarne uno. Per ora supponete di voler usare il seguente prefisso: **fd00:dead:bad:1dea::/64**.
Per semplicità configurerete inizialmente l'identificatore dell'host (il resto dell'indirizzo del router IPv6) staticamente, a **::1** per brevità. Il comando **ip** è un buon metodo per farlo:

```
$ sudo ip addr add dev eth0 fd00:dead:bad:1dea::1/64
```

Dove **eth0** è il nome dell'interfaccia della vostra rete (la vostra potrebbe essere diversa). Ora potete fare ping a voi stessi su IPv6:

```
$ ping6 -c 5 fd00:dead:bad:1dea::1
```

Per rendere persistente l'assegnamento di indirizzo tra i reboot dovete capire il modo della vostra distribuzione: può essere fatto con la GUI di **NetworkManager** o con i vari script di networking (per esempio **netctl** su Arch Linux). Se volete solamente una rete IPv6, andate oltre, altrimenti leggete di seguito. Per comunicare con il resto del mondo parlante IPv6 (e assumendo che il vostro ISP non vi fornisca già tale connettività) dovete inviare il vostro traffico tramite IPv4 usando il popolare protocollo 6in4. Un'opzione consiste nel registrarvi a un servizio di »

Esercizio di astensione da IPv4

L'avvento di IPv6 è stato, nonostante l'interessamento di alcuni grandi nomi e sia obbligatorio per lo standard mobile 4G, notoriamente lento. Parte della ragione di ciò sta nel fatto che IPv4 si è rivelato un campione. Le estensioni di sicurezza IPSec e DNSSec, originariamente pensate come parte di IPv6, vi sono state incorporate, togliendo una ragione per il passaggio. Nel 2014, tuttavia, molti router aziendali meno recenti hanno cominciato a impazzire via via che le tabelle BGP eccedevano i 512 k elementi. Si specula che sia la causa di alcuni downtime di eBay e Lastpass. Sebbene tutti abbiano recuperato piuttosto in fretta, è stato un campanello d'allarme per gli interessati. Quasi all'unisono diverse parti del sistema hanno cominciato a scricchiolare e gli amministratori di rete sono passati all'azione, con il risultato che ora, se spegnete IPv4, la maggior parte di Internet continua a funzionare come vi aspettate. Se state usando un tunnel Teredo e lo avete impostato per comunicare sulla porta UDP 3544, le seguenti regole di iptables bloccheranno ogni altra richiesta in uscita, tagliando effettivamente tutte le connessioni IPv4 a eccezione del vostro server DNS:

```
# iptables -A OUTPUT -m state --state
ESTABLISHED,RELATED -j ACCEPT
# iptables -A OUTPUT -p UDP --dport 53 -j
ACCEPT
# iptables -A OUTPUT -p UDP --sport 3544 -j
ACCEPT
# iptables -A OUTPUT -j REJECT
```

Troverete alcuni siti funzionanti, altri più o meno, la maggior parte no. Probabilmente la connettività IPv4 vi servirà ancora, quindi una volta terminato l'esperimento dovreste eliminare queste regole. Se non ne avevate applicata alcuna in precedenza, è questione di un semplice:

```
# iptables -F
```

```
ciromattia@mercurio-ubuntu: ~
ciromattia@mercurio-ubuntu:~$ sudo systemctl start miredo
ciromattia@mercurio-ubuntu:~$ routel
         target            gateway          source    proto    scope    dev tbl
        default           10.0.2.2                   static            eth0
      10.0.2.0 24                                    kernel     link   eth0
   169.254.0.0 16                                               link   eth0
       10.0.2.0          broadcast       10.0.2.15   kernel     link   eth0 local
      10.0.2.15              local       10.0.2.15   kernel     host   eth0 local
     10.0.2.255          broadcast       10.0.2.15   kernel     link   eth0 local
      127.0.0.0          broadcast       127.0.0.1   kernel     link     lo local
    127.0.0.0 8              local       127.0.0.1   kernel     host     lo local
      127.0.0.1              local       127.0.0.1   kernel     host     lo local
127.255.255.255          broadcast       127.0.0.1   kernel     link     lo local
       2001:: 32                                     kernel          teredo
fd00:dead:bad:1dea:: 64                                   kernel           eth0
       fe80:: 64                                     kernel            eth0
       fe80:: 64                                     kernel          teredo
        default                                                      teredo
        default        unreachable                   kernel              lo unspec
            ::1              local                     none              lo local
2001:0:53aa:64c:3883:2e6e:af26:a284          local                  none              lo local
fd00:dead:bad:1dea::1              local                  none              lo local
fe80::ffff:ffff:ffff              local                  none              lo local
fe80::a00:27ff:fe0a:345              local                  none              lo local
        ff00:: 8                                                       eth0 local
        ff00:: 8                                                     teredo local
        default        unreachable                   kernel              lo unspec
ciromattia@mercurio-ubuntu:~$ ping6 -c 2 ipv6.google.com
PING ipv6.google.com(arn06s07-in-x0e.1e100.net) 56 data bytes
64 bytes from arn06s07-in-x0e.1e100.net: icmp_seq=1 ttl=55 time=160 ms
64 bytes from arn06s07-in-x0e.1e100.net: icmp_seq=2 ttl=55 time=49.1 ms

--- ipv6.google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 49.167/104.867/160.567/55.700 ms
ciromattia@mercurio-ubuntu:~$
```

› I tunnel Teredo sono l'ultima spiaggia per ottenere un tunnel diretto IPv6. Non dovreste farlo, anche se aggiunge un sapore da frutto proibito. Specialmente se non potete mettere mano alla configurazione del vostro router...

» intermediazione tunnel come Hurricane Electric o SixXS. Offrono tale servizio gratuitamente ma dovete registrarvi, cosa che per SixXS comprende due approvazioni umane. Ci sono diversi modi di incapsulare il traffico IPv6 in pacchetti IPv4, ma se siete dietro un layer NAT la soluzione più semplice è il protocollo *Anything in Anything* (AYIYA). SixXS ha fornito la comoda utility *Automatic IPv6 Connectivity Configuration Utility* (disponibile nel pacchetto **aiccu**) che rende facile configurare il tunnel. Alcune distro vi chiedono le credenziali SixXS nell'installazione, mentre in altre dovrete modificare manualmente **/etc/aiccu.conf** così che assomigli a questo:

```
username <username>
password <password>
protocol tic
server tic.sixxs.net
IPv6_interface sixxs
automatic true
requiretls true
pidfile /var/run/aiccu.pid
defaultroute true
makebeats true
behindnat true
```

Quando lanciate **aiccu** dovreste essere a posto.

Hurricane Electric (**https://tunnelbroker.net**) richiede di specificare manualmente i capi del tunnel (tramite la sua interfaccia Web, dove scegliete un endpoint geograficamente vicino a voi, $HE_ENDPOINT4 nell'esempio sotto), e di configurare manualmente il tunnel a lato client. Inoltre la compagnia richiede che l'IP esterno del vostro router sia pingabile dal mondo esterno, anche se non è richiesto il port forwarding. I messaggi ICMP sono necessari per negoziare le route IPv6. Hurricane Electric vi assegna un indirizzo IPv6 per l'endpoint client che inizia con **2001:** e termina con **/64**, che chiamerete $CLI_IP6 nel codice sotto. Siete liberi di impostare tanti host quanti ve ne pare in questa rete; non c'è bisogno di impostare un nuovo tunnel per ognuno di questi. Il tunnel è impostato come segue, dove $INT_IP4 è il vostro indirizzo IP interno. Notate che questi comandi devono essere eseguiti da root, quindi usate sudo se necessario:

```
# ip tunnel add he-IPv6 mode sit remote $HE_
ENDPOINT4 local $INT_IP4 ttl 255
# ip link set he-IPv6 up
# ip addr add $CLI_IP6 dev he-IPv6
# ip route add ::/0 dev he-IPv6 # ip -f inet6 addr
```

Dovrete inoltrare il traffico del protocollo 41 (non della porta 41) sul vostro router, anche se molti non ve lo permetteranno (ma non i router su cui gira DD-WRT). Se vi sentite coraggiosi potete mettere la vostra macchina nella DMZ del router così da farvi redirigere tutte le connessioni in entrata. È un grosso rischio di sicurezza, comunque, quindi non fatelo. Se non riuscite a farlo funzionare, seguite le istruzioni per Teredo.

Dovreste sapere che la funzionalità DNS del vostro router potrebbe non ritornare record IPv6. Questo può essere aggirato usando i server DNS di Google, aggiungendo **nameserver 8.8.8.8** a **/etc/resolv.conf** o modificando le impostazioni del Server DNS in NetworkManager. Le query DNS su IPv4 possono comunque ritornare indirizzi IPv6, ma potete anche usare gli indirizzi IPv6, in questo caso **2001:4860:4860::8888**. Provatelo con:

```
$ ping6 -c 5 IPv6.google.com
```



› Buone notizie: apparentemente siete abili e arruolati! In effetti però non è proprio così...

Presentazione del router

Oltre a impostare manualmente un indirizzo IPv6 statico o farvi assegnare un prefisso da un broker, come visto prima, potete ottenere un indirizzo IPv6 in altri due modi. Il primo è la *Stateless Address Auto Configuration* (SLAAC), dove un host interpella un router tramite il Neighbor Discovery Protocol. Un router locale risponde con un prefisso di rete e altri parametri di configurazione. Combinando tutto ciò con l'identificatore dell'host, usando la derivazione dal MAC o le estensioni per la privacy descritte più avanti, ottenete un indirizzo IPv6. Potete anche usare il più tradizionale approccio stateful tramite DHCPv6, dove il server tiene traccia di quali indirizzi ha assegnato a chi. Per permettere ai client della vostra rete di ottenere informazioni IPv6, il vostro router deve rispondere alle richieste con una presentazione. Senza questa, nessuno sentirebbe i vagiti

Tunnel Teredo

Come alternativa all'uso di un tunnel broker per una singola macchina, è possibile usare un tunnel **Teredo**. Teredo è una tecnica pensata da Microsoft che si basa sull'ampiezza di banda di diversi relay Teredo. Questi ultimi funzionano da gateway, disincapsulando i vostri pacchetti IPv6 e inviandoli, e racchiudendoli in pacchetti IPv4 in ritorno. È intesa come ultima spiaggia, e dal momento che un solo indirizzo IPv6 può essere assegnato per ogni tunnel, non è adatta all'uso su reti, anche piccole. Dal momento che crea un tunnel che può tranquillamente bypassare il vostro firewall, è ampiamente riconosciuta come rischio di sicurezza. In generale, il suo utilizzo tende a essere scoraggiato. Tuttavia è molto buona per attraversare i vari layer di un NAT (**teredo** è una famiglia di molluschi in grado di forare gli scafi delle navi) quindi torna utile se il vostro indirizzo IPv4 esterno cambia regolarmente o se la vostra rete non vi aiuta. Un client Open Source chiamato **Miredo** è disponibile tra i pacchetti della vostra distribuzione. Una volta installato e lanciato il servizio (**systemctl start miredo** per Systemd) dovreste godere di IPv6. Gli indirizzi Teredo hanno un prefisso **/32** riservato e tutti iniziano con **2001:0:**. Se tutto va come pianificato, dovreste averne uno assegnato. Potete provarlo visitando **http://test-IPv6.com** che vi darà (a voi a alla vostra connettività) un voto da 1 a 10. Non avrete sempre il massimo con una connessione Teredo ed è anche possibile che il servizio Miredo generi un IP interno errato. In quest'ultimo caso modificate semplicemente **/etc/miredo/miredo.com** e aggiungete una riga simile a

```
BindAddress 192.168.1.10
```

delle vostre interfacce senza indirizzo. Il demone per l'advertising si trova nel pacchetto **radvd** che dovrete installare. Creerà una semplice rete IPv6 locale senza connettività esterna, anche se è possibile presentarsi anche al tunnel intermediato della sezione precedente. il file **/etc/radvd.conf** d'esempio contiene molte definizioni di esempio. Per mantenere le cose semplici userete questo semplice setup:

```
mv /etc/radvd.conf{,.example}
# nano /etc/radvd.conf
```

Ora dovete aggiungere alla configurazione, sostituendo **eth0** con il nome della vostra interfaccia:

```
interface eth0 {
        AdvSendAdvert on;
        MinRtrAdvInterval 3;
        MaxRtrAdvInterval 10;
        prefix fd00:dead:bad:1dea::/64 {
                AdvOnLink on;
                AdvAutonomous on;
                AdvRouterAddr on;
        };
};
```

Se avete impostato un tunnel Hurricane Electric potete usare il prefisso **/64** assegnato. Questo permette alla vostra rete IPv6 di vedere il mondo esterno. Dovrete abilitare l'inoltro del traffico IPv6 per farlo funzionare:

```
# echo 1 > /proc/sys/net/ipv6/conf/eth0/forwarding
```

Ora lanciate il servizio con

```
$ sudo service radvd start
```

oppure con

```
# systemctl start radvd
```

a seconda del vostro sistema di init. Se va tutto bene, connettere altre macchine nella rete ora le vedrà auto-configurarsi con SLAAC e come bonus avrete configurato la vostra macchina locale. Per renderlo persistente, abilitate il servizio **radvd** ($ systemctl enable radvd) e cambiate l'impostazione

```
net.IPv6.conf.default.forwarding=1
```

in **/etc/sysctl.conf** (o un file tipo **/etc/sysctl.d/10-ip6-forward.conf**). Il vostro router si presenta a sé stesso, quindi lanciare questo servizio vi configura altresì un indirizzo IPv6. Il comportamento standard è di generare un identificatore di host dal MAC come descritto prima, ma alcune distribuzioni hanno attivato le estensioni privacy per default, quindi verrà generato un identificatore casuale (solitamente una volta al giorno). Controllate come sono impostati i client (ancora sostituendo **eth0** con la vostra interfaccia):

```
$ cat /proc/sys/net/IPv6/conf/int0/use_tempaddr
```

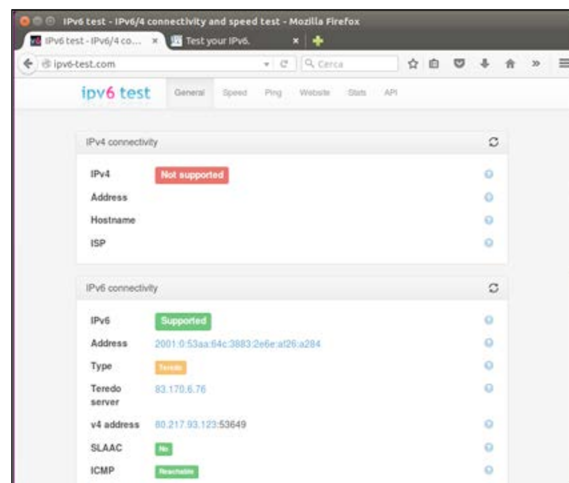
Se ritorna 0, le estensioni privacy non sono abilitate. Per rimediare, come root, lanciate

```
# echo 2 > /proc/sys/net/ipv6/conf/eth0/use_tempaddr
```

Come prima, le impostazioni possono essere rese persistenti aggiungendo

```
net.IPv6.conf.int0.use_tempaddr = 2
```

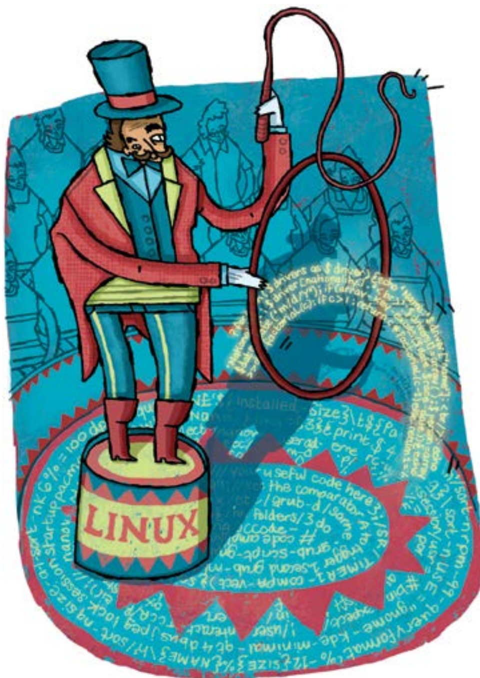
nel file **/etc/sysctl.d/20-ip6-tempaddr.conf**. Ricordate che una volta ottenuto un indirizzo IPv6 (tramite il vostro ISP, un intermediario o Teredo) la vostra macchina diventa accessibile al mondo esterno, così come qualsiasi servizio in esecuzione su questa. Con IPv6 la dipendenza da NAT non esiste più, e allo stesso modo tante frustrazioni nell'impostazione dei client BitTorrent. È prudente impostare un qualche genere di firewall, ma fintantoché i vostri servizi sono configurati per ascoltare solamente interfacce IPv4 dovreste essere a posto. Un effetto collaterale piuttosto intimorente di IPv6 è la nascita di reti ombra, che appaiono quando il traffico IPv6 riesce a evadere le misure di sicurezza impostate solamente per IPv4, permettendo a un attaccante di aggirare casualmente qualsiasi politica del firewall. In un futuro prossimo, si spera, gli ISP forniranno connettività IPv6, rendendo questo tutorial obsoleto. LXP



› Ci sono piccole ricompense tangibili nel negare a voi stessi i piaceri offerti da *IPv4*, ma il 6 giugno 2014 è proprio quello che molti sysadmin hanno fatto

Costruire un gioco di parole

Non sapete scrivere codice? Non avete voglia di farlo? Lo staff di Linux Pro vi insegnerà come creare un intero gioco con una manciata di righe di PHP



Cinque anni fa Linux Mint era la distribuzione per i radical-chic, **Systemd** stava nascendo e Linux sembrava prendere sempre più piede nei computer di tutto il mondo. Oggi, Systemd rappresenta una delle più grandi controversie per la comunità Linux, l'esplosione di utilizzatori deve ancora arrivare e per quanto riguarda Linux Mint? Ok, Linux Mint è ancora abbastanza gettonato, uno su tre, non male. In questo tutorial andremo a vedere alcuni elementi di base del PHP imparando a costruire un gioco di parole da riga di comando. Il gioco chiederà ai concorrenti di scrivere delle parole utilizzando dei gruppi di lettere, come il gioco per smartphone **7 Little Words**. Per esempio, i gruppi 'SIG', 'ARE', 'TTA' formano la parola 'SIGARETTA' che risponde all'indizio 'Più la tiri più si accorcia'. Per semplificare le cose, il gioco dirà al concorrente il numero di lettere di ogni risposta. Non preoccupatevi se non siete molto esperti di programmazione: nel corso del tutorial cercheremo di spiegarvi tutto ciò che lo spazio a disposizione ci permetterà, in questo modo anche i principianti potranno venirne a capo. Prima di partire ovviamente dovete installare **PHP**. Se state utilizzando una distribuzione derivata da Debian vi basterà utilizzare questo comando `apt-get install php5-cli`. Le altre distribuzioni offrono tutte dei pacchetti simili, ma assicuratevi di installare la versione di PHP da riga di comando invece del modulo **Apache**, se il vostro gestore di pacchetti vi dice che dovete installare Apache probabilmente avete sbagliato qualcosa. Questo però è solo il primo passo, ora dovrete installare un pacchetto che vi permetta di leggere gli input degli utenti da riga di comando. Non è presente di default tra i pacchetti Debian, purtroppo, ma è molto semplice aggiungerlo. Per prima cosa lanciate

```
sudo nano /etc/apt/sources.list
```

e in fondo, dopo tutto il resto, aggiungete:

```
deb http://packages.dotdeb.org wheezy all
```

Se state utilizzando una debian basata su Jessie o Squeeze potete mettere `jessie` o `squeeze` al posto di `wheezy` e funzionerà correttamente. Salvate il file e uscite da **Nano** (**Ctrl**+**o**, **Invio**, e **Ctrl**+**x**) e lanciate questi comandi:

```
wget http://www.dotdeb.org/dotdeb.gpg
sudo apt-key add dotdeb.gpg
sudo apt-get update
sudo apt-get install php5-readline
```

Questo aggiornerà la vostra installazione di PHP permettendole di accedere alla riga di comando. Si potrebbe pensare che questo sia il minimo da un pacchetto chiamato PHP-CLI (*PHP Command Line Interface* (interfaccia da linea di comando)), ma evidentemente non è così.

Un altro passo

Prima di iniziare a scrivere codice, dobbiamo soffermarci a pensare a come saranno i nostri livelli. Ogni livello avrà sette indizi, e ogni indizio avrà due o tre gruppi di lettere che compongono la risposta. Per evitare di complicare troppo il nostro progetto non utilizzeremo l'XML, ma un formato molto più semplice:

1. Ogni livello avrà sette righe.
2. Ogni riga conterrà i suoi gruppi di lettere, due punti, e il suo indizio.
3. I gruppi di lettere saranno divisi dal pipe: |.

Tenendo presente quanto detto, ecco come si presenta un file di livello, salvatelo nel vostro computer con il nome di **livello1.txt**:

```
MU|US|EO: Il posto preferito di una mucca
SCO|TCH: Attaccante americano
FUR|GON|CIN: Il ministro dei trasporti cinese
OT|TOV|AS|CHE: Campione di nuoto tedesco
```

```
SIG|ARE|TTA: Più la tiri più si accorcia
AS|SUN|TA: Nome di disoccupata
SAL|SA: Lo fa il pomodoro la mattina
```

Una volta che avrete finito di soffrire per i terribili giochi di parole, spremete le meningi per creare altri file, chiamateli **livello2.txt**, **livello3.txt** e via di seguito, li aggiungeremo al nostro codice in seguito. Qualsiasi cosa scriviate, assicuratevi di concludere con 20 gruppi per rendere l'interfaccia armoniosa. Questo formato è pensato per essere facile da leggere, facile da modificare e facile da processare per PHP, per esempio per trasformare il testo in un gioco reale. Il fatto che sia facile per voi da leggere e da modificare è molto importante, perché ogni volta che leggete questi terribili giochi di parole perdete un po' di intelletto. Il fatto che sia facile da leggere per PHP è altrettanto importante, perché possiamo fargli caricare i dati con soltanto una manciata di righe di codice. Non ci credete? Vediamo come!

Leggere i dati con PHP

Ogni linguaggio di programmazione Unix degno di questo nome è bravo nell'elaborazione dei testi, e PHP non fa differenza: potete caricare un file semplicemente con una riga di codice, potete dividere le righe utilizzando i due punti e i pipe con un'altra riga di codice, e potete rimuovere tutti gli spazi bianchi in eccesso con un'altra riga di codice. Anche se non è ai livelli di Perl, PHP è decisamente un linguaggio conciso. Per prima cosa scriveremo una funzione chiamata **caricaLivello()**. In seguito andrà chiamata passandole il nome del file da caricare (**livello1.txt** è il file che abbiamo appena creato), e lei si occuperà di trovare gli indizi e le soluzioni all'interno del file e di organizzare tutti i dati. All'interno della nostra funzione andremo a creare tre array, o raccolte di valori. Il primo conterrà tutti i gruppi di lettere, per esempio, 'MU', 'US', 'EO', 'SCO', 'TCH', eccetera. Il secondo conterrà tutti gli indizi, ma per farlo dovrà contenere tre dati, l'indizio in se ('Più la tiri più si accorcia'), l'aiuto che dice di quante lettere è formata la risposta ('9 lettere'), e i gruppi di lettere che formano la risposta. Quest'ultimo valore è necessario per permetterci di rimuovere dallo schermo i gruppi utilizzati quando il giocatore risolve un enigma. Il terzo array salverà soltanto le risposte per ogni livello in modo da poterle trovare facilmente.

Prima di tuffarci nel codice, ci sono sei punti fondamentali sul PHP che bisogna ricordare:

1 Tutti i valori iniziano con $; anche se non rappresentano del denaro, ma se volete impressionare i vostri amici sappiate che il simbolo **$** utilizzato per le variabili è chiamato sigillo.

2 Potete creare un array vuoto scrivendo due parentesi quadre in questo modo: **[]**. Potete utilizzarle anche per aggiungere un valore a un array. Gli array in PHP partono da zero, perché… beh, in realtà non c'è un motivo.

3 Potete leggere ogni oggetto di un array utilizzando un ciclo. Andremo a utilizzare tre tipi diversi di cicli, non lo faremo soltanto per crearvi confusione, i cicli sono tutti utili, davvero!

4 La funzione **file()** carica un file in un array nel quale ogni riga del file corrisponde a un valore dell'array. È perfetto per i nostri scopi, ma fate attenzione: lascia un'interruzione di riga alla fine di ogni valore, quindi dovrete rimuoverla.

5 La funzione **explode()** converte del testo in un array suddividendolo in base a uno specifico carattere.

La utilizzeremo per suddividere i dati in base ai due punti, perché la parte a sinistra contiene i gruppi di lettere e quella a destra l'indizio, poi la useremo di nuovo per suddividere i gruppi di parole che sono separati tra loro dal simbolo |.

6 Quando una funzione deve tornare un valore al codice che l'ha invocata è necessario scrivere **return** seguito dal valore che si vuole ritornare.

Esistono un altro paio di funzioni utili, ma hanno nomi abbastanza chiari, quindi dovreste riuscire a capire cosa fanno guardando il codice. Scrivete il codice che segue in un file chiamato **parole.php** e salvatelo sul vostro computer vicino a **livello1.txt**:

```
<?php
function caricaLivello($nomefile) {
        $gruppi = [];
        $indizi = [];
        $soluzioni = [];

        $input = file("livello1.txt");

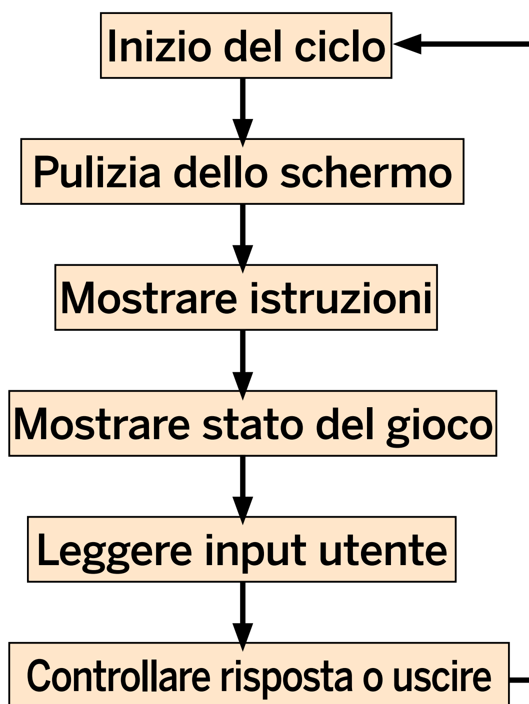
        foreach ($input as $riga) {
                $riga = explode(":", $riga);
                $parti = explode("|", $riga[0]);
                $gruppi = array_merge($gruppi,$parti);

                $soluzione = str_replace("|","",$riga[0]);
                $soluzioni[] = $soluzione;
                $indizi[] = ["indizio" => trim($riga[1]),
"lunghezza" => strlen($soluzione). " lettere","parti" => $parti];
        }
        shuffle($gruppi);
        return ["lettere" => $gruppi, "soluzioni" =>
$soluzioni, "indizi" => $indizi];
}
```

Una volta salvato il file, potete eseguirlo da riga di comando con `php parole.php` sfortunatamente non farà nulla.

Vi sembrerà un po' demoralizzante scrivere tutto quel codice e non vedere alcun risultato, ma tranquillizzatevi, arriverà presto. Per prima cosa, vediamo di spiegare le altre funzioni utilizzate:

» **array_merge()** unisce insieme due array. Lo utilizziamo per »



› Utilizziamo un ciclo infinito in questo modo il gioco non esce finché l'utente non scrive *quit* o chiude il suo terminale

» unire insieme tutti i gruppi di lettere.

» **str_replace()** sostituisce una parte di testo all'interno di una variabile. Nel nostro caso, sostituiamo il simbolo pipe (il |) con un valore vuoto. Il codice **$riga[0]** significa 'il primo valore dell'array', ecco cosa intendevamo quando abbiamo detto che gli array in PHP partono da zero.

» **trim()** rimuove tutti gli spazi vuoti in eccesso dal testo. È necessario perché la funzione **file()** lascia le interruzioni di riga alla fine dei dati che legge.

» **strlen()** misura la lunghezza di un testo.

» **shuffle()** mescola casualmente un array, rendendo il gioco un po' più interessante.

La sola cosa complicata nella funzione caricaLivello() è il modo in cui vengono utilizzati gli array, per esempio la funzione ritorna questo:

```
return ["lettere" => $gruppi, "soluzioni" => $soluzioni, "indizi" => $indizi];
```

Abbiamo già detto che gli array contengono un insieme di valori, ma ci sono due modi in cui possono farlo. Nel primo i dati sono salvati in ordine numerico, come nel caso di $riga[0] e $riga[1]. Nel secondo, che è quello che potete vedere qui sopra, invece di utilizzare le posizioni 0, 1, 2, e via di seguito, diamo un nome a queste posizioni. In questo modo l'array che ritorniamo può essere letto utilizzando $array["lettere"] o $array["soluzioni"], che è più facile da ricordare. Se ve lo stavate chiedendo il <?php significa 'tutto da qui è codice PHP, e va trattato come tale'. Potete uscire dalla modalità PHP con ?>, tutto ciò che verrà dopo sarà stampato a video.

Lo stato di avanzamento

La funzione caricaLivello() prepara i dati necessari al nostro gioco, ma non contiene alcuna logica. Abbiamo bisogno ancora di:

1. Mostrare lo stato corrente del gioco. Questo deve mostrare tutti i gruppi di lettere e tutti gli indizi. Oltre agli indizi dobbiamo mostrare da quante lettere è composta la risposta se non è ancora stata scoperta, oppure la risposta corretta quando viene indovinata.
2. Chiedere all'utente di indovinare finché il gioco non è completato.
3. Quando l'utente indovina, rimuovere i gruppi di lettere utilizzati e aggiornare gli indizi per mostrare la risposta corretta.

La prima cosa da fare è mostrare lo stato attuale del gioco: tutti i gruppi di lettere e tutti gli indizi. Lo faremo grazie a una nuova funzione chiamata stampaStato() che si aspetta due parametri: l'array di gruppi di lettere da mostrare e il nostro array con gli indizi. Per stampare i gruppi di lettere andremo a utilizzare un secondo tipo di ciclo chiamato ciclo for, esso ci permette di contare all'interno di un range di numeri, che nel nostro caso è tra lo 0 e il numero di gruppi che abbiamo. Utilizzeremo anche una tecnica chiamata modulo, conosciuta anche come 'resto della divisione'. Un'operazione di modulo significa 'che valore rimane quando dividiamo un numero X per un numero Y?' per esempio 20 modulo 5 è 0 perché il 5 sta esattamente 4 volte nel 20 e non c'è resto, mentre 22 modulo 5 corrisponde a 2 perché il 5 sta 4 volte nel 22 e rimane un resto di 2. Il modulo è utile per la nostra funzione perché vogliamo mostrare le lettere in modo ordinato. Insieme al ciclo for, conteremo anche da 0 fino al numero di gruppi di lettere (che è 20), e mostreremo ogni gruppo sullo schermo seguito da un carattere tab per allontanarli l'uno dall'altro. Quindi, utilizzando il modulo, ogni volta il numero corrente nel nostro ciclo si divide per 5 con un resto di 4, aggiungeremo due interruzioni di riga. In pratica, questo creerà una griglia di gruppi composta da cinque colonne e quattro righe. Stampare gli indizi è molto più semplice: utilizzeremo un secondo ciclo for per contare da 0 a 7, leggendo l''indizio' e la 'lunghezza' per ogni indizio. Abbiamo bisogno di altre tre cose prima di procedere con il codice:

1. Per aggiungere 1 a un numero possiamo scrivere $vostroNumero, che significa 'aggiungi uno a questo numero e salva il risultato nella stessa variabile'.
2. Se volete aggiungere un tab nel testo potete utilizzare il \t. Per creare una nuova riga invece \n.
3. Accedere agli array all'interno del testo da stampare può creare confusione, quindi PHP preferisce che si utilizzino le parentesi graffe. Per esempio: "Ecco il valore: {$array["mioValore"]}".

Ora il codice, inserite quanto segue nel vostro script PHP sotto alla funzione caricaLivello():

```
function stampaStato($lettere, $indizi) {
        for ($i = 0; $i < count($lettere); ++$i) {
                echo "{$lettere[$i]}\t";

                if ($i % 5 == 4) {
                        echo "\n\n";
                }
        }

        echo "\nINDIZI\n";

        for ($i = 0; $i < 7; ++$i) {
                echo $i + 1, ".{$indizi[$i]["indizio"]}
({$indizi[$i]["lunghezza"]})\n";
        }

        echo "\nInserisci la risposta:\n";
}
```

La riga di codice più complessa è all'interno del ciclo degli indizi, perché deve trovare informazioni specifiche all'interno di un grande array. $indizi[$i]["lunghezza"], significa 'prendi l'array $indizi, trova l'oggetto alla posizione $i, e recupera il valore della lunghezza', ma fa tutto ciò in una sola riga.

Ora, abbiamo una buona e una cattiva notizia. La cattiva notizia è che se eseguite questo script, non avrete alcun risultato. Ma la buona notizia è che manca pochissimo prima che tutti gli ingranaggi vadano al loro posto, perché ciò accada abbiamo bisogno di un altro tipo di ciclo... Fin qui abbiamo scritto due funzioni, ma sono le due che formano la spina dorsale del nostro programma. Per rendere il nostro gioco operativo, dobbiamo eseguire la funzione caricaLivello() o e salvare quello che ci ritorna e poi iniziare a chiedere ai nostri utenti di indovinare. Perché questo accada, abbiamo bisogno di un nuovo tipo di ciclo chiamato ciclo infinito. È un ciclo che rimane in esecuzione per sempre, o almeno finché qualcuno non lo ferma ed è perfetto per un gioco nel quale l'utente continua a fare tentativi finché non vince. All'interno di questo ciclo dovremo pulire lo schermo, stampare le istruzioni e lo stato del gioco, leggere l'input del giocatore e utilizzarlo. Parliamo dell'input dell'utente, per prima cosa andremo

Variabili in stringhe

Il PHP fa un largo uso di una tecnica chiamata interpolazione delle stringhe, che significa che potete scrivere 'Ciao $variabile bla bla' e $variabile verrà rimpiazzato dal suo valore all'interno del testo. Questo ovviamente è molto utile, ma ha i suoi problemi: come fa PHP a sapere dove finiscono le variabili e inizia il testo? Nell'esempio precedente è chiaro: le variabili PHP non possono avere spazi bianchi nel nome, quindi PHP vede lo spazio dopo $variabile e sa che lì finisce la variabile. Quando si parla di array, dovreste usare $variabile["dato"], ma significa il contenuto della variabile $variabile seguito dal testo ["dato"], oppure il contenuto di "dato" nell'array $variabile? PHP offre una soluzione semplice a questo problema, racchiudere il tutto tra parentesi graffe: { e }. Non è richiesto per cose semplici come $variabile, ma lo è nel nostro codice perché utilizziamo dati come $indizi[$i]["indizio"] ed è molto ambiguo. Quando vi trovate in questa situazione, mettete semplicemente delle parentesi attorno alle vostre variabili in questo modo: {$indizi[$i]["indizio"]}, e le parentesi chiariranno esattamente cosa volete dire.

a togliere gli spazi vuoti dall'inizio e dalla fine e lo convertiremo in lettere maiuscole. È importante perché non ci interessa se l'utente scrive 'sigaretta', 'SIGARETTA' o qualsiasi combinazione nel mezzo. Fatto ciò controlleremo due casi particolari: se l'input è vuoto (per esempio l'utente ha premuto soltanto l'**Invio**) continuiamo il nostro ciclo dall'inizio, e se l'input è **Q** o **QUIT** (sempre maiuscolo ovviamente), usciamo dal programma. Per rendere le cose un po' più semplici da leggere, questo nuovo pezzo di codice suddividerà il valore ritornato da caricaLivello() in tre valori separati.
Per il computer non è necessario, ma è importante per mantenere il codice leggibile, fidatevi!
Inserite questo codice alla fine del vostro script PHP:

```
$livelloCorrente = caricaLivello("livello1.txt");
$lettere = $livelloCorrente["lettere"];
$soluzioni = $livelloCorrente["soluzioni"];
$indizi = $livelloCorrente["indizi"];

while (true) {
        system("clear");
        echo "\nLE SETTE PAROLE DI LINUX\n";
        echo "Dovrete scrivere sette parole utilizzando i
gruppi di lettere qui sotto.\n";
        echo "Per giocare, provate a scrivere una parola
formata da due o più gruppi.\n";
        echo "Potete uscire in qualunque momento
scrivendo 'quit'.\n\n";
        stampaStato($lettere,$indizi);

        $riga = strtoupper(trim(readline()));
        if ($riga == "") continue;

        if ($riga == "QUIT" || $riga == "Q") {
                echo "Grazie per aver giocato!\n";
                exit;
        }
        // controllare la risposta
}
```

Ci sono diverse cose che necessitano di una spiegazione.
Per prima cosa, il ciclo infinito si ottiene con il comando while (true), che si traduce in 'finché vero è vero, esegui questo codice'. Ovviamente vero è sempre vero, quindi il ciclo continua per sempre. Poi, per pulire lo schermo del terminale abbiamo usato la funzione system() con il parametro 'clear'. Questo simula che l'utente abbia scritto 'clear' sul terminale, è una buona scorciatoia. Terza cosa, la funzione strtoupper() è nuova, ma è molto semplice: converte tutto il testo in maiuscolo e lo ritorna. Quarto, anche la funzione readline() è nuova, ma si spiega da sola: legge l'input dell'utente. Quinto, i due pipe || all'interno dell'if significano 'oppure', quindi questa riga significa 'se l'input dell'utente è Q oppure è QUIT ...'
Per finire, verso la fine c'è una riga che dice // controllare la risposta. Il PHP ignora tutto ciò che inizia con '//', perché è un commento.

Controllare le risposte

Se eseguite il codice ora, vedrete che il programma inizia a funzionare. Ma il gioco non è ancora veramente un gioco, potete scrivere tutto quello che volete ma non succederà nulla di interessante. Fortunatamente siete a pochissimi minuti dalla conclusione del vostro gioco. Tutto ciò che resta da fare è sostituire quel commento // controllare la risposta con il codice che controlla se la risposta dell'utente è valida. Se lo è, dobbiamo rimuovere i gruppi di lettere dal gioco, aggiornare l'elenco degli indizi per mostrare la risposta, e togliere quella risposta da quelle possibili. Per fare tutto ciò andremo a utilizzare una nuova funzione chiamata array_search() che ci dice se un elemento esiste all'interno di un array e dove si trova. Abbiamo già il nostro array $soluzioni, che contiene tutte le possibili risposte, quindi useremo array_search() per cercare il testo inserito in input dall'utente in questo array.
E ora una scorciatoia: la posizione della risposta nel nostro array $soluzioni è la stessa dell'indizio corrispondente nell'array degli indizi. Per aggiornare la lista degli indizi per mostrare la risposta invece che la lunghezza dobbiamo cambiare l'indizio alla stessa posizione della risposta.
Poi per rimuovere i gruppi di lettere utilizziamo array_search() per trovare ogni gruppo all'interno dell'array $lettere, e lo impostiamo come testo vuoto in modo che non sia più visibile:

```
$posizione = array_search($riga, $soluzioni);

if ($posizione !== false) {
        $indizi[$posizione]["lunghezza"] =
$soluzioni[$posizione];

        foreach ($indizi[$posizione]["parti"] as $parteLettere)
{
                $posizioneLettere = array_
search($parteLettere, $lettere);
                if ($posizioneLettere !== false) {
                        $lettere[$posizioneLettere] = "
";
                }
        }

        $soluzioni[$posizione]='';
}
```

E questo è tutto! Il vostro gioco è finito, potete giocare. LXP

Tip

Ora avete un gioco completo tra le vostre mani. Ma c'è ancora molto che potete fare. Per prima cosa potreste fare un altro livello, poi modificare il codice per cambiare livello quando il giocatore indovina tutte le parole. Fatto questo, provate a salvare i punteggi: aggiungete 10 per ogni risposta corretta, e togliete 1 per ogni errore.

Gestire il sistema: i processi

In questa quinta puntata esplorerete i processi **fork()**, **exec()** e compagnia, e capirete cosa intende un programmatore per **pipe**

Ecco un piccolo esperimento per voi, come quella cosa di Schrödinger e del gatto (che non vi consigliamo di fare, a meno che non vogliate una visita dagli animalisti). Immaginate di poter smantellare il vostro computer fino a isolarne i singoli componenti. Finireste con una pila di cavi, connettori, viti, resistenze e condensatori. Quello che non vi rimarrebbe è sicuramente una pila di processi. Il che è strano, perché se prima di aprire la vostra macchina aveste lanciato il comando

```
$ ps -e | wc -l
```

avreste scoperto che ne esistevano più di 100 nella vostra macchina. Quindi, esattamente, cos'è un processo? E come si crea? Un processo è un concetto astratto. Una definizione comune è 'un'istanza di un programma in esecuzione', anche se non è di grandissimo aiuto. Potete però pensare a un processo anche come un tipo di contenitore che mantiene il contesto necessario a un programma per girare.
Per aggiungere un po' di sale, la tabella sottostante mostra alcuni degli attributi più importanti di un processo.

Attributo	Descrizione
Segmento codice	L'area di memoria che contiene il codice eseguibile
Segmento dati	L'area di memoria per le variabili globali e altri dati statici
Stack	Area di dimensione variabile che contiene variabili locali
Heap	Memoria usata per l'allocazione dinamica dell'archivio
ID processo	Identificativo univoco per il processo
ID utente reale	Identificativo dell'utente che sta eseguendo il processo
ID utente effettivo	L'utente con i cui permessi sta girando il processo
Directory corrente	Da dove il processo inizierà a calcolare i percorsi relativi
Variabili d'ambiente	Lista di coppie NOME=VALORE usate per la personalizzazione
File aperti	Lo stato (per esempio il puntatore del file) di tutti i file aperti
Disposizioni dei segnali	Come il processo gestirà i vari tipi di segnale

Tip

Potete scovare facilmente tutte le named pipe nel vostro sistema con il comando **$ sudo find / -type p**.

Negli articoli precedenti avete già incontrato alcuni di questi attributi e ne vedrete altri nei prossimi mesi. I processi sono creati con la chiamata di sistema fork(). Sintatticamente, fork() è la chiamata di sistema più semplice di tutte: non prende alcun argomento e ritorna semplicemente un intero come risultato. Ma da un altro punto di vista è la più strana di tutte, dal momento che anche se un solo processo effettua la chiamata, da essa ritornano due processi. Confusi? Un'analogia potrebbe aiutarvi. Esattamente come un programma è una lista di istruzioni di quello che c'è da fare, il copione di un'opera teatrale è come una lista di istruzioni per quello che c'è da dire e fare. Estendendo l'analogia, un attore è come un processo: è l'agente responsabile per eseguire effettivamente le istruzioni. Potreste dire che il programma è passivo, ma il processo è attivo.

Pensate ai bambini!

Quindi il nostro attore è da solo sul palco e legge l'istruzione "fork()", a questo punto chiama un altro attore dalle quinte per affiancarlo sul palco. Il secondo attore ha una copia identica del copione e infatti, leggono insieme la riga seguente a fork(). Sono processi separati ma entrambi, almeno inizialmente, eseguono lo stesso programma. L'analogia non è perfetta. Il nuovo processo non sta realmente aspettando dietro le quinte, pre-creato e pronto all'azione: viene invece creato dalla chiamata a fork(). Il processo originale viene definito genitore (*parent*), quello nuovo il figlio (*child*) e per cominciare è una copia pressoché esatta del genitore. C'è tuttavia un'importante differenza che permette ai due processi di capire chi è chi dopo il ritorno di fork(): nel genitore, fork() ritorna l'ID del processo figlio appena creato, mentre nel figlio fork() ritorna zero. Ciò significa che vedrete sempre, invariabilmente, la chiamata fork() in un test simile a questo:

```
if (fork()) {
        /* sono il genitore */
} else {
        /* sono il figlio */
}
```

Ecco un esempio più corposo nel quale genitore e figlio eseguono un loop stampando dei messaggi:

```
#include <stdio.h>
```

Perché forkare?

Perché i programmi scelgono di forkare? Esistono due ragioni principali. La prima è per ottenere concorrenza all'interno di una singola applicazione. Per esempio un server Web, mail o database può scegliere di forkare per servire più client contemporaneamente. Vero è che il multi-threading (non discusso qui) potrebbe offrire una soluzione più robusta ed efficiente; tuttavia il modello processo-figlio-per-client rimane popolare. La seconda ragione è la delega. L'idea è che vogliate delegare un certo compito a un altro programma, ma volete mantenerne il controllo così da sapere quando è terminato per proseguire. Questo è esattamente l'ambito di tutto il ciclo fork/exec/exit/wait, ed è quello che la shell esegue ogni volta che inserite un comando. Se volete elencare una directory, la shell delega il compito a ls eseguendo (con exec) tale programma, ma lo fa in un processo child separato così da poterne mantenere il controllo e chiedervi il comando seguente.

```
void main()
{
        int i;
        if (fork()) {
                for (i=0; i<100000; i++)
                        printf("** GENITORE %d\n", i);
        } else {
                for (i=0; i<100000; i++)
                        printf("** FIGLIO %d\n", i);
        }
}
```

Sarebbe spazio sprecato mostrarvi l'output, quindi scrivetelo e provatelo direttamente! Troverete che le righe GENITORE e FIGLIO sono alternate in maniera imprevedibile, dal momento che lo scheduler divide il tempo della CPU tra i due processi. Se lanciate di nuovo il programma l'alternanza sarà diversa. Può spaventare pensare che anche l'output di questo minuscolo programma sia non-deterministico, ma è quello che ottenete quando cominciate a introdurre la concorrenza nel mix. In teoria potreste anche vedere l'output alternato in una singola riga, così:

```
** GE ** FIGLIO 5NITORE 547713880
```

In pratica è molto difficile vedere un cambio di contesto all'interno di una riga, ma il principio rimane: i due processi stanno girando contemporaneamente e l'ordine in cui accadranno le cose è casuale. Talvolta, lanciando questo programma, il prompt seguente della shell non vi comparirà. In effetti non è sparito ma semplicemente più in alto dell'ultimo output: questo succede se il figlio termina dopo il genitore, dal momento che anche se la shell aspetta che il genitore finisca prima di stampare il prompt, non aspetta il figlio. Vedrete questa cosa più avanti. Talvolta genitore e figlio continuano a eseguire lo stesso programma (vedete il box **Perché forkare?**), ma spesso il figlio è destinato a eseguire un programma completamente diverso. Lo fa con una chiamata della famiglia che, se volete un nome di gruppo, chiamerete exec(). Tornando all'analogia con l'attore e il copione, una exec() è come un'istruzione nel copione che indica "Vai e recita il copione di Macbeth". Quindi l'attore butta via il copione che sta attualmente leggendo, prende una copia di Macbeth, la apre a pagina 1 e comincia a leggere "A quando riunirci noi tre nuovamente? Tra tuoni, lampi, o nella pioggia battente?" È lo stesso attore, semplicemente sta leggendo un copione diverso.

Le sette versioni di exec()

Come mostrato nel diagramma decisionale sulla destra, esistono sette versioni di exec(). I sette nomi sono oscuri, ma seguono una regola comune:

1 se il nome termina con 'e' viene passato a un nuovo ambiente, altrimenti viene mantenuto quello vecchio;

2 se il nome ha una 'p' all'interno, l'eseguibile viene cercato nel percorso di ricerca, altrimenti dovrete passare un percorso assoluto;

3 se il nome ha una 'l' all'interno, gli argomenti da riga di comando vengono passati come elenco esplicito nella chiamata, altrimenti se ha una 'v' vengono passati come vettore (array).

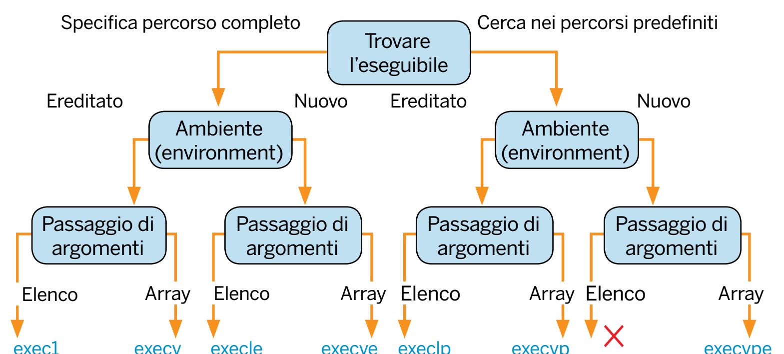
Alcuni esempi potrebbero aiutarvi:

```
char *argv[] = {"ls", "-l", NULL};
execl("/bin/ls", "ls", "-l", NULL);
execv("/bin/ls", argv);
execlp("ls", "ls", "-l", NULL);
execvp("ls", argv);
```

Notate che una chiamata exec() eseguita con successo non ritorna. Il processo ora sta eseguendo qualche altro programma. Se exec() ritorna, c'è stato un errore, probabilmente perché non è stato possibile trovare l'eseguibile o non c'erano permessi sufficienti a eseguirlo.

Terminazione di processi

Questo è quanto per la creazione dei processi. Come disfarsene? Il modo più comune è che il programma termini volontariamente chiamando exit(). Questa chiamata prende come argomento un intero che viene restituito al genitore ed è definito exit status. Per convenzione un exit status uguale a zero indica successo e non-zero (1-255) implica qualche genere di errore. Quello che viene considerato fallimento dipende dal programma, per esempio in **grep** il fallimento è non trovare una corrispondenza per il pattern, nel qual caso viene restituito un valore pari a 1. Questa convenzione confonde i programmatori C, che si aspettano 0 pari a falso e non-zero uguale a vero. Il genitore può scegliere di aspettare l'uscita del figlio chiamando wait(&status) dove status è una variabile intera. La chiamata blocca l'esecuzione finché uno »



› **Le sette variazioni di *exec()* possono intimorire. Questo diagramma di flusso vi aiuterà a scegliere quella che fa per voi**

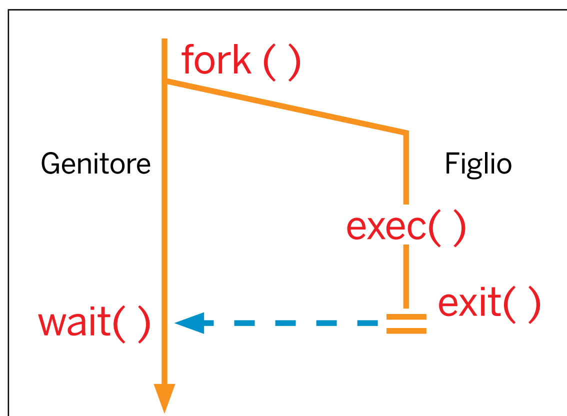
Tip

Nei moderni kernel Linux la dimensione predefinita di una pipe è 65.536 byte. C'è tuttavia un limite molto inferiore (4.096 byte) sulla quantità di dati per cui è garantita una scrittura atomica (cioè tutta in una volta, senza paura di avere delle interferenze con qualcosa scritto da un altro processo). Questo è da tenere a mente se avete diversi processi che scrivono nella stessa pipe.

qualsiasi dei processi figlio esce (ricordate che potreste aver forkato diverse volte quindi avere più figli). Il PID del figlio uscito viene ritornato come valore della funzione. L'exit status viene ritornato nel byte alto della variabile status e potete analizzarlo usando la macro WEXITSTATUS. Ora sapete come scrivere una semplice shell. Davvero! Eccola:

```
1. #include <stdio.h>
2. #include <string.h>
3. #include <stdlib.h>
4.
5. int prompt_and_parse(char **args)
6. {
7.         static char line[100];
8.         printf("> ");
9.         if (gets(line) == NULL)
10.                return -1;
11.        *args++ = strtok(line, " \t");
12.        while (*args++ = strtok(NULL, " \t"))
13.                /* Vuoto */ ;
14.        return 1;
15. }
16.
17. main()
18. {
19.        char *args[20];
20.
21.        /* Ciclo principale per i comandi */
22.        while (prompt_and_parse(args) > 0) {
23.                if (fork() == 0) { /* Figlio */
24.                        execvp(args[0], args);
25.                        printf("%s: non trovato\n",
args[0]);
26.                        exit(1);
27.                }
28.                else { /* Genitore */
29.                        wait(0);
30.                }
31.        }
32. }
```

Ecco un veloce tour guidato. La funzione prompt_and_parse (righe 5-15) richiede un comando, legge una riga e la separa in token, inserendoli nell'array passato come argomento. Attenzione: il buffer a lunghezza fissa alla riga 7 e l'uso di gets() aprono il fianco ad attacchi di buffer overflow, quindi non dovreste usarlo in produzione (allo stesso modo il buffer fisso alla riga 19 non è una buona pratica). Il ciclo per il controllo della stringa alla riga 12 è una di quelle cose che amate o odiate. Tutto il lavoro effettivo viene fatto come effetto collaterale della valutazione della condizione di test per il while(). Il ciclo di comando principale (righe 22-31) effettua chiamate ripetute a prompt_and_parse(), forkando ogni volta e facendo eseguire al figlio la riga di comando appena processata. Notate la scelta di execvp() in questo caso. Alla riga 29 il genitore aspetta semplicemente che il figlio abbia terminato (ignorando l'exit status) prima di ricominciare il ciclo. Notate anche la gestione dell'errore alla riga 25. Se execvp() fallisce, presumibilmente perché non riesce a trovare l'eseguibile, viene attivato questo blocco: provatelo. Il prompt originale $ è della shell (Bash) originale, gli altri di **Smallshell**.



› Le quattro chiamate fork(), exec(), exit() e wait() sono i mattoni per il ciclo di vita del processo

```
$ smallshell
> date
Wed May 13 19:50:59 BST 2015
> date -I
2015-05-13
> file smallshell.c
smallshell.c: C source, ASCII text
> ls -l *.c
ls: cannot access *.c: No such file or directory
> cd ..
cd: non trovato
```

Ebbene, i primi tre comandi hanno funzionato correttamente, quindi il parsing della riga di comando sembra essere OK. Il comando seguente va male perché la vostra shell non espande le wildcard. Il tentativo finale con cd, infine, fallisce (fallisce proprio execvp()) perché cd non è un comando esterno e dev'essere integrato nella shell stessa. Ma seriamente, cosa vi aspettavate da 30 righe di codice?

Non capirci un tubo

Tutti sanno cosa sono le pipe (in italiano, tubi). Sono i meccanismi di comunicazione inter-processo più ampiamente usati in Linux e sono al cuore della filosofia classica della costruzione di strumenti che sfruttano combinazioni di programmi, come:

```
$ ps -e | wc -l
```

Ora guardate più da vicino le pipe con gli occhi di un programmatore di sistema. Le pipe sono unidirezionali. Hanno un capo upstream nel quale potete scrivere e un capo downstream da cui potete leggere. La pipe ha una dimensione finita e impone una sincronizzazione tra un produttore e un consumatore: il produttore si bloccherà nel tentare di scrivere su una pipe piena, mentre il consumatore si bloccherà nel leggere una pipe vuota. Le pipe vengono create con la chiamata pipe():

```
int p[2];
pipe(p);
```

Vi vengono restituiti due file descriptor: p[0] nel capo downstream e p[1] nel capo upstream. Ora, per sfruttare una pipe per comunicare tra due processi fate affidamento su due cose: in primo luogo, i descrittori sono ereditati da un processo figlio durante una fork(); in secondo luogo, i descrittori sopravvivono a una chiamata exec(); ciò significa che se, per esempio, il file descriptor 4 è aperto nell'upstream di una pipe prima di una exec(), sarà aperto nel nuovo programma dopo la exec() (in realtà è una semplificazione: potete esplicitamente impostare la flag close-on-exec su un file descriptor per evitarlo, ma non lo affronterete qui). Creare e usare una pipe può essere diviso in quattro stadi (vedete pagina seguente):

```
int p[2];
pipe(p);
if (fork()) {
        /* Genitore */
        dup2(p[0], 0);
        close(p[1]);
        exec( ... downstream ...);
} else {
        /* Figlio */
        dup2(p[1], 0);
        close(p[0]);
        exec( ... upstream ...);
}
```

Ecco la spiegazione. Allo stadio 1, il processo padre (A) crea una pipe, ricevendone i descrittori di entrambi i capi. Allo stadio 2 il processo forka e il processo figlio (B) eredita i descrittori. Allo stadio 3, il processo genitore (A), che in questo esempio è destinato a diventare il processo downstream, copia il capo downstream della pipe nel proprio standard input. Quindi chiude il proprio descrittore sul capo upstream. Questo risulta essere piuttosto importante, dal momento che quando il processo downstream cercherà in seguito di leggere dalla pipe, si bloccherà se la pipe è vuota fintantoché almeno un processo avrà un descrittore aperto nel capo upstream.
Ora, naturalmente, il processo A non ha intenzione di scrivere alcunché nella pipe, ma è il mero fatto di tenere ancora aperto il descrittore che causa il problema. Allo stadio 4, il processo figlio duplica il capo upstream della pipe nel suo standard output e chiude il capo downstream. Ora è tutto impostato: il processo B può scrivere nella pipe e il processo A può leggere dalla questa. La decisione di rendere il processo genitore il downstream e il figlio upstream è arbitraria: potrebbe anche essere il contrario. In effetti è molto più comune avere il genitore che forka due volte e configura i due figli come processi upstream e downstream.
Ma questo esempio serve solamente a illustrare come funzionano le pipe. L'altro punto importante è che normalmente i processi figli non hanno idea di cos'è stato fatto dal genitore per manipolare i propri descrittori di file, semplicemente si limitano a leggere il proprio standard input o scrivere nello standard output. Nella maggior parte dei meccanismi di comunicazione inter-processo c'è bisogno di un 'punto di ritrovo' pre-concordato per permettere a un processo di trovare l'altro. Nel caso di un socket TCP, per esempio, il client deve conoscere l'indirizzo IP e il numero di porta su cui il server sta ascoltando. Come altro esempio, per una coda di messaggi in stile System-V tutti i processi devono concordare sull'ID della coda. Le pipe che avete visto tuttavia non hanno nome o identificatore e vengono talvolta definite pipe anonime. Funzionano perché i due processi comunicanti sono in relazione, avendo un antenato comune che ha creato la pipe e passato loro il descrittore.

Jack la pipe

Va da sé quindi arrivare alle named pipe. Funzionano come le pipe anonime ma sono visibili nel filesystem, permettendo a processi non relazionati di comunicare, fintantoché è noto il nome della pipe. Dalla riga di comando potete creare una named pipe così:

```
$ mkfifo /tmp/miapipe
```

o, all'interno di un programma, potete semplicemente usare la chiamata `mkfifo()`:

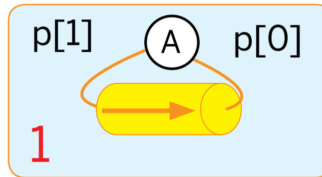
```
mkfifo("/tmp/miapipe", 0666);
```

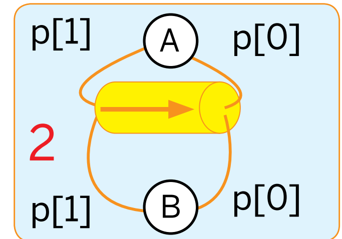
Nel caso ve lo steste chiedendo, fifo sta per *first in, first out* (primo arrivato, primo servito), che è la caratteristica chiave dell'accesso alle pipe. Una named pipe ha un'esistenza permanente nel filesystem. Appariranno come tipo 'p' se effettuate un listato lungo:

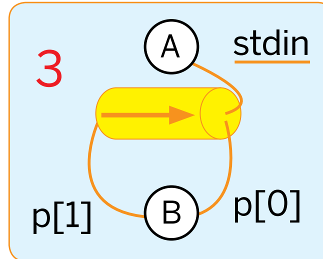
```
$ mkfifo /tmp/miapipe
$ ls -l /tmp/miapipe
prw-rw-r-- 1 lestat lestat 0 May 19 18:25 /tmp/miapipe
```

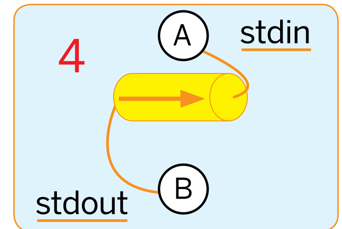
D'altra parte, una pipe anonima esiste solo finché c'è almeno un processo con un suo descrittore aperto. Dal punto di vista della programmazione di sistema, le named pipe vengono aperte, lette e scritte esattamente come un file, eccetto che per risultare utili dovete avere un processo che le abbia aperte in lettura e un altro in scrittura. Quindi, invece di vedere altro codice C, ecco un piccolo esperimento che potete provare con le named pipe da riga di comando:

1 Aprite due finestre di terminale (le chiamerete A e B).
2 Nella finestra A create la pipe **/tmp/miapipe**, come appena indicato.
3 Nella finestra A lanciate il comando `$ cat /tmp/miapipe`. Si bloccherà perché `cat` resterà in attesa di qualche altro processo che apra la pipe in scrittura.
» Nella finestra B lanciate il comando `$ cat > /tmp/miapipe`.
» Ora scrivete qualche riga di testo nella finestra B. Mano a mano che cat le scrive nella pipe, verranno prese dal cat in esecuzione nella finestra A e stampate.
» Inserite `$ ^D` (**Ctrl**+**D**) nella finestra B: questo terminerà il cat in upstream. Quando succede, il cat in downstream (finestra A) riceverà un 'end of file' quando tenterà di leggere dalla pipe (perché nessun altro ha il capo upstream aperto) e terminerà, ritornandovi alla prompt dei comandi. Per questo mese è tutto. Al prossimo con l'identità dei processi e il controllo di accesso! LXP









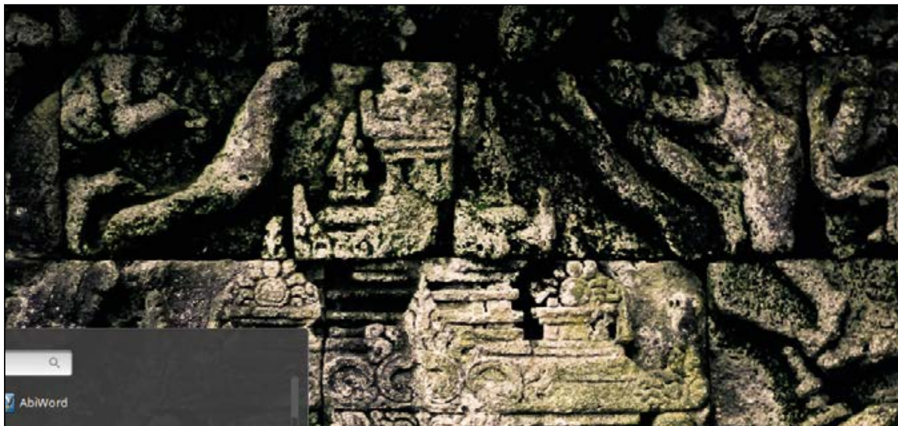
› I quattro stadi per impostare una pipe. Leggete il testo per una spiegazione completa

Domande & Risposte

Avete qualche domanda sull'Open Source? Scriveteci a **mailserver@linuxpro.it** per ottenere una risposta

1 Dov'è il mio backup?

Vorrei aggiornare la versione 14 di Linux Mint alla 17.1. Purtroppo, però, ho dei problemi a eseguire il backup dei miei file dal disco fisso a una chiave USB da 8 GB Cruzer. La procedura sembrava essere andata a buon fine, ma quando ho controllato nella chiave USB, i file non c'erano. Per l'occasione ho provato a usare anche un software specializzato nel backup, ma il risultato è stato lo stesso.



› ***Linux Mint* è disponibile in versione 17.2 *Rafaela*, ma Clement Lefebvre ha già annunciato la 17.3 che sarà chiamata *Rosa***

Qualsiasi programma di backup tu abbia usato, dovrebbe permetterti di scegliere un percorso in cui salvare i tuoi file. Tuttavia esiste un modo molto semplice per trovare un documento mancante, anche se non ti ricordi come si chiama. Basta usare il comando **find**.

```
$ find /media/chiaveusb -type f -mtime -1
```

Questa istruzione consente di reperire e stampare i nomi di tutti i file modificati negli ultimi giorni. Se vuoi eseguire una ricerca ancora più ristretta nel tempo, usa:

```
$ touch somefile
```

quindi avvia il programma di backup con:

```
$ find /media/chiaveusb -type f -newer somefile
```

Il comando **touch** imposta la marca temporale (timestamp) sul file desiderato per l'ora corrente, mentre **find** controlla tutti i file più recenti rispetto a quello creato. Se ancora non riesci a trovare il tuo backup, allora passa a crearne uno tramite il terminale. La procedura è abbastanza semplice:

```
$ tar cf /media/chiaveusb/home-backup.tar ~
$ tar czf /media/chiaveusb/home-backup.tar.gz ~
```

Uno di questi due comandi provvederà a copiare la directory **home** in un file compresso all'interno della chiave USB. Utilizzando la compressione, tieni presente che il processo di salvataggio potrebbe richiedere un po' di tempo, soprattutto se i dati da archiviare occupano parecchio spazio. Per decomprimere l'archivio nella stessa directory, usa il comando:

```
$ tar xf /media/usbstick/home-backup.tar.gz
```

Ricorda che potresti comunque incorrere in un potenziale problema. A seconda del programma di backup che stai utilizzando, la chiave USB potrebbe essere formattata in FAT. Questo filesystem non è in grado di salvare documenti che occupano più di 4 GB. Il modo più semplice per risolvere, è riformattare l'unità con il filesystem **ext2**. Puoi farlo sia con i tradizionali strumenti messi a disposizione dalla tua distro, sia lavorando da riga di comando con la seguente istruzione:

```
$ sudo mke2fs /dev/sdXN
```

dove sdXN è il dispositivo USB. In futuro

Riflettori su...

Il termine Software Libero significa gratuito?

No, non è così. La gratuità di un programma e la sua appartenenza al gruppo del Software Libero sono due condizioni completamente diverse. Con questo termine ci si riferisce esclusivamente alla licenza d'uso con cui un'applicazione viene distribuita. Per rientrare nella branca del Software Libero, un programma deve garantire almeno quattro dei seguenti punti:

01 - libertà di eseguire il programma per qualsiasi scopo;
02 - libertà di studiare il programma e modificarlo;
03 - libertà di ridistribuire copie del programma;
04 - libertà di migliorare il programma e di distribuirne pubblicamente i miglioramenti.

In definitiva, un Software Libero può essere anche a pagamento.

Terminale e superutente

Nelle distro Linux ci sono diversi modi per applicare un comando a una data operazione. La maggior parte delle volte, infatti, potete procedere sia utilizzando degli strumenti grafici sia la potenza della riga di comando. In quest'ultimo caso, avrete a disposizione strumenti molto più flessibili e pratici da usare, ma soprattutto validi per tutte le distribuzioni. Prendendo come esempio i comandi per la configurazione di sistema, avrete sicuramente notato come nel momento in cui tentate di eseguirne uno, vi venga richiesto l'accesso come *superuser* o *root*. In Ubuntu e nelle sue derivate, per esempio, questo è identificato con il comando **sudo** a cui poi deve seguire l'inserimento della password. Altre distro come Fedora, invece, utilizzano **su**, che danno accesso all'utente root fino a quando non si inserisce il comando **logout**.

è comunque possibile evitare questi problemi, creando una partizione separata per **/home**. Puoi farlo direttamente dal processo d'installazione del sistema operativo. In questo modo, nel caso si voglia procedere a un aggiornamento della piattaforma o alla sua formattazione, i tuoi dati non verranno toccati. Passare da Mint 14 a 17.1 senza reinstallare può essere abbastanza azzardato. Di solito si esegue il semplice aggiornamento senza formattazione quando si scala di una singola versione, ma qui stiamo parlando di ben tre release. Molto meglio formattare e reinstallare da capo.

2 Problemi con Dracut

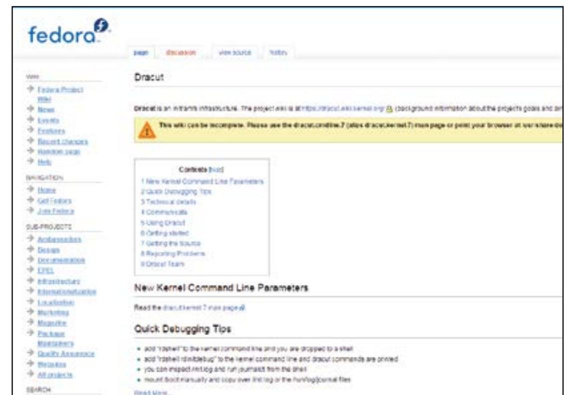
Sto tentando di installare Dracut, ma ho riscontrato diversi problemi. Il programma si è rivelato un incubo di dipendenze contrastanti e pacchetti corrotti. Per quanto sia riuscito a capire, l'unico modo per installarlo è rimuovere initramfs e una lista infinita di file. I pacchetti per cui mancano le dipendenze sono i seguenti:

```
dracut : Conflicts: initramfs-tools
but 0.103ubuntu4.2 is installed.
Conflicts: initramfs-tools:i386
```

quindi l'errore continua con:

```
Keep the following packages at
their current version:
1) dracut [Not Installed]
```

Il pacchetto **initramfs-tools** contiene i programmi/script attualmente utilizzati da Linux Mint per generare i propri file initramfs. Questi vengono poi usati da altri pacchetti per ricostruire gli initramfs dopo l'installazione di alcuni aggiornamenti. Gli sviluppatori hanno quindi deciso di mettere in conflitto Dracut con gli initramfs perché il programma potrebbe sovrascriverli mandando in crash il sistema. Il problema, dal nostro punto di vista, è più di natura etica che tecnica. Il punto di forza della filosofia del Software Libero consiste proprio nel permetterci di modificare qualsiasi cosa, anche se questo significa danneggiarla. Dall'altra parte, anche il pensiero dei programmatori di Dracut è condivisibile. Se con il loro software ci consentono liberamente di danneggiare il sistema, non torneremo certo a utilizzarlo. Niente impedisce di disinstallare initramfs-tools, ma per una semplice prova di Dracut ci sembra un'operazione troppo radicale. La soluzione è comunque semplice e a portata di mano: installa Dracut dai sorgenti, così da bypassare i controlli effettuati dal gestore pacchetti. In questo modo, potrai avere sia Dracut sia gli initramfs-tolls montati nel sistema, senza che il gestore pacchetti sia cosciente della presenza di Dracut. Ricorda che il programma può tentare di creare dei file initramfs con lo stesso nome di quelli originali, ma non li sovrascriverà a meno di non utilizzare il comando **--force**. Se dovesse verificarsi questa eventualità, specifica a Dracut di usare un nome diverso (si può fare dalla riga di comando), oppure rinomina i file originali. In entrambi i casi sarà necessario creare le apposite voci nel menu di avvio di Grub, così da utilizzare sia gli initramfs originali sia Dracut. Tieni comunque a mente che questa attività potrebbe influire su altre distribuzioni presenti.



› **Per tutte le informazioni su Dracut, collegatevi alla wiki di Fedora: https://fedoraproject.org/wiki/Dracut**

3 Come usare SSMTP

Sto usando Fedora 21 e non posso gestire le mie email. Ho installato SSMTP, ma ancora non riesco a far lavorare la funzione php mail(). Credo di dover modificare il file di configurazione di SSMTP, ma non so dove si trovi. Cosa posso fare?

SSMTP è un *forwarder* che viene scambiato per *sendmail* dai programmi che lo interpellano. In realtà, tutto quello che fa è dirottare la posta su un altro server. Questo significa che i programmi »



Riferimento rapido

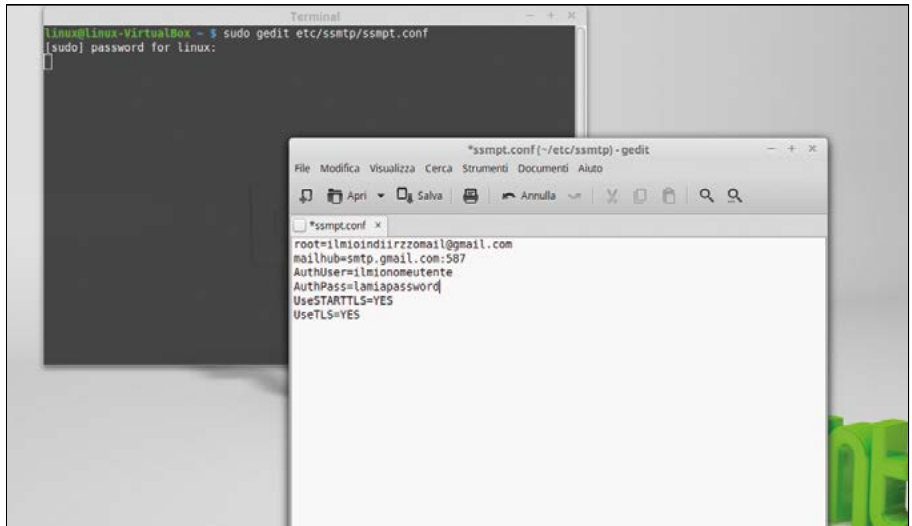
Top

Sicuramente vi sarete imbattuti in un computer lento, che non risponde ai comandi o che non ne vuol sapere di funzionare velocemente. In questo caso, provate a eseguire il comando **top** dal terminale. Questo programma consente di reperire un'ingente quantità di informazioni sullo stato del PC. Nella parte superiore, trovate le statistiche di sistema che riportano il numero di utenti, le operazioni in esecuzione e via dicendo. Di seguito vengono riepilogati i dettagli riguardanti il processore: l'utilizzo complessivo, la percentuale usata dai programmi utente, quella riferita ai servizi di sistema, ecc. Scorrendo la pagina, trovate l'elenco delle attività che impegnano maggiormente la CPU. La lista è disposta per esecuzione, quindi le prime voci presenti sono quelle che pesano di più sull'uso del processore. Per cambiare ordinamento, è sufficiente premere il tasto **F**. Per esempio, si può visionare al volo quale programma occupa più memoria. Per terminare un processo, basta utilizzare il comando **k** seguito dall'ID del processo (PID) che viene mostrato nella colonna di sinistra. Fate comunque attenzione agli errori di battitura. Potrebbero essere fatali.

per la gestione della email possono inviare messaggi senza utilizzare un server di posta installato nel computer. Il file di configurazione di SSMTP è il seguente: ***/etc/ssmtp/ssmtp.conf***. Prima di tutto è però necessario modificarlo in alcune voci. L'opzione più importante a cui mettere mano è **mailhub**. Si tratta dell'indirizzo del server di posta utilizzato. Può essere in locale o sulla rete LAN, ma più in generale sarà quello del server messo a disposizione dal provider. Se l'host di posta utilizza una porta diversa da quella standard (25), è possibile specificarlo in mailhub:

```
$ mailhub=mail.tuo.dominio:2525
```

A tal proposito, potrebbe essere necessario inserire l'hostname che determina quello usato da SSMTP durante l'invio delle mail. Ricorda che se questo parametro non corrisponde a quello utilizzato dall'indirizzo di posta, alcuni server potrebbero dare problemi. Se non configurato, viene comunque utilizzato l'hostname predefinito del computer (riportato da **hostname -f**). Se l'host di posta o il provider utilizzano l'autenticazione SMTP, è poi necessario configurare le voci **AuthUser** e **AuthPass** con le credenziali di accesso corrette.



› Il file di configurazione di SSMTP può essere aperto con il comando *gedit etc/ssmtp/ssmtp.conf*

Potrai comunque trovare tutti i dettagli riguardanti la configurazione del file **ssmtp.conf** nella pagina man corrispondente.

4 Provare Linux

Vorrei installare una versione di Linux su un disco fisso secondario. In più, gradirei che si avviasse premendo il tasto F8 in fase di accensione del PC. La distro dovrebbe poi riconoscere le altre unità (C, D, E, ecc.), così da poter gestire i vari documenti in modalità multi-piattaforma. Il mio PC monta Windows 7 Professional, un processore octa-core AMD e 16 GB di RAM. In più, ho sei dischi interni. Per installare Linux devo forse scollegare tutte le unità tranne quella in cui monto la distribuzione?

La soluzione del mese

Copia e decompressione

Sono nel pieno di un trasferimento dati tra il mio vecchio disco fisso e uno nuovo appena comprato. Ho però un problema. Dispongo di parecchi documenti che iniziano con la sigla IT seguita da una cifra, una parola tra parentesi e l'estensione del file. In pratica, sono rinominati in questo modo: *IT 1 (Nome File).txt*. Il guaio consiste nel fatto che il comando *cp* non digerisce le parentesi. La classica operazione di copia con il tasto destro del mouse sui file non funziona. Devo quindi operare da terminale con *sudo*. I documenti in questione erano contenuti in uno ZIP che ho decompresso con il comando:

```
$ unzip file.zip -d /la-directory-in-cui-voglio-memorizzarli
```

A questo punto, però, anziché collocare i documenti nella directory che volevo, è stata creata una sottocartella. Sicuramente ho sbagliato qualcosa, ma non capisco dove. Potreste spiegarmi bene il procedimento da seguire?

Il problema consiste nel fatto che il sistema considera le parentesi come caratteri speciali utilizzati per il pattern matching. Questa procedura consiste nel controllare la presenza di un certo elemento (il pattern appunto) all'interno di un oggetto. La shell, quindi, tenta di interpretare le parentesi come comandi specifici e non come appartenenti al nome di un file. Se tu provassi a usare **ls** anziché **cp**, il risultato sarebbe lo stesso. Per fare in modo che la shell digerisca correttamente i tuoi documenti, è opportuno eliminare sia i caratteri speciali (le parentesi) sia gli spazi. Per farti un esempio, dovresti rinominare i file così:

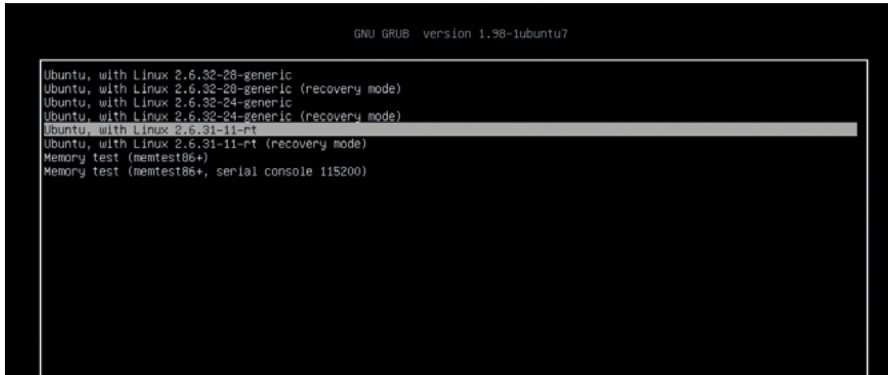
```
IT\ 1\ \(Nome\ File\).txt
IT\ 2\ \(Magna\ Carta\).txt
```

Puoi fare in modo che la shell aggiunga i caratteri di escape utilizzando la scheda di completamento. Prova per esempio **Uni[TAB]1[TAB]**. Così facendo, però, il rischio di incorrere in errori di battitura è piuttosto alto. Potresti quindi risolvere utilizzando un carattere jolly per copiare direttamente tutti i file:

```
$ sudo cp -p Unit[TAB]*.txt /destination/path
```

Per quanto riguarda lo ZIP, è importante ricordare che si tratta di un archivio di file con una propria directory strutturata. Quando si decomprime, quindi, la struttura rimane la stessa. La funzione **-d** consente di eseguire la decompressione della struttura in questione nel percorso indicato anziché nella directory corrente. Se invece i file sono memorizzati all'interno di una sola cartella (senza sotto-cartelle), è possibile usare l'opzione **-j**. In questo modo, tutti i file saranno decompressi in una directory di destinazione, a prescindere dalla loro posizione nell'archivio. Se invece si vuole mantenere la struttura, bisogna usare questa serie di comandi:

```
$ sudo unzip -d /dest/path/tuoarchivio.zip
$ sudo mv /dest/path/tuoarchiviodir/* /dest/path
$ sudo rmdir /dest/path/tuoarchiviodir
```



› Ecco un esempio di un menu per l'avvio dei sistemi operativi generato da Grub

Quasi tutte le distro Linux possono fare quello che chiedi, ma è meglio scegliere una delle più popolari. L'installer della distribuzione dovrebbe poi raccogliere le unità di Windows e montarle all'avvio di Linux, così da averle disponibili in un percorso simile a **/mnt/windows_C**, **/mnt/windows_D** e via dicendo.
La posizione esatta può variare, ma sarà comunque facile da reperire nel file manager. Visto che usi Windows 7, i dischi saranno sicuramente formattati in NTFS. Questo filesystem ha però un limite nei confronti di Linux: consente la sola lettura dei dati, ma non la scrittura. Per risolvere, usa il gestore pacchetti della distro e installa **ntfs-3g**. In questo modo, sarai in grado di scrivere sulle partizioni NTFS anche da Linux. Per l'installazione non c'è assolutamente bisogno di scollegare gli altri dischi fissi. Qualsiasi installer di una distro rileva le varie unità, quindi ti chiede dove installare la distribuzione. A questo punto potrai seguire due strade: installare un bootloader sul disco extra e scegliere di avviare il sistema dal menu di boot del PC (come richiesto), oppure lasciare che il programma d'installazione di Linux monti il bootloader sull'unità primaria del sistema. In questo caso, ti verrà proposto un menu da cui scegliere quale piattaforma lanciare tra Windows e Linux. Di solito, se non si effettua una scelta entro 10 secondi, il computer avvierà il sistema predefinito. Scegliendo questo approccio, c'è però una considerazione importante da fare. Nel caso in cui si rimuova Linux, il menu di avvio scomparirà. Per risolvere basta inserire il CD d'installazione di Windows e, tramite la console di ripristino, utilizzare il comando **fixmbr**. In alternativa, procurati il CD di **Rescatux** che ripristinerà il bootloader di Windows al volo. LXP

Soluzione rapida

La quantità di distro a disposizione degli amanti del mondo Linux è davvero enorme. Non passa giorno in cui non vengano rilasciate nuove versioni. Per chi si avvicina per la prima volta al Pinguino, può non essere facile scegliere quella più adatta per le proprie competenze e per l'uso che se ne intende fare. Un primo consiglio che diamo, è dare un'occhiata a **Distrowatch**, **www.distrowatch.com**. In seconda battuta, potete fare un semplice test che vi dirà quale distro è più adatta a voi. Lo trovate alla pagina **www.linux.it/scegli**. In linea generale, comunque, le migliori distribuzioni per i principianti sono Ubuntu e Linux Mint.







FAQ: le domande più frequenti

WPA

› Cosa significa WPA?
WPA è il successore del vecchio protocollo di sicurezza WEP.

› Grazie per questa spiegazione concisa, ma potresti essere più preciso?
WPA, acronimo di *Wi-Fi Protected Access*, è un sistema che consente di garantire la privacy dei dati trasmessi attraverso una rete wireless.

› Ne ho bisogno?
A differenza di una tradizionale rete cablata, dove i dati non lasciano mai la sicurezza dei cavi che collegano il router ai PC, in una rete Wi-Fi la trasmissione avviene in ogni direzione e si estende per centinaia di metri. Ciò significa che chiunque con un computer portatile e una scheda wireless potrebbe captare le informazioni che transitano senza difficoltà.

› La mia casa è enorme, quindi dovrei essere al sicuro, giusto?
Sbagliato! Basta aggiungere un comune extender per potenziare la trasmissione del segnale. Quindi è sufficiente che qualcuno capti anche una minima traccia per poterla ottimizzare e poi sfruttare per carpire i dati.

› Che cos'è WEP?
WEP è l'acronimo di *Wired Equivalent Privacy*. Si tratta di uno dei primi sistemi utilizzati per proteggere le reti wireless. Purtroppo, con il tempo, si è rivelato poco affidabile. Infatti si può bypassare con delle tecniche piuttosto semplici.

› Per sfruttare WPA devo disporre di hardware specifico?
Tutte le schede di rete wireless costruite recentemente supportano sia la modalità WPA sia WEP. Se però il tuo PC monta una periferica piuttosto vecchia, potrebbe non riuscire a gestire WPA ma solo WEP.

› Sul fronte software ho bisogno di qualche programma?
In questo caso il supporto WPA deve essere esteso su due livelli: nei driver della periferica e in un programma separato per l'implementazione di WPA. Quello più popolare è **wpa-supplicant** che dovrebbe comunque essere disponibile per tutte le distro. In alternativa, lo trovi su **http://hostap.epitest.fi/wpa_supplicant**. Questo software gestisce la negoziazione delle chiavi e la relativa autenticazione.



L'eco dei LUG

I Lug

I LUG rappresentano da sempre il punto di riferimento per chiunque voglia conoscere GNU/Linux. Ogni mese dedicheremo loro questo spazio per la comunicazione di nuovi progetti e appuntamenti. Se hai qualcosa da segnalarci scrivi a **ecodeilug@linuxpro.it**

ABRUZZO

AnxaLUG - Lanciano
www.anxalug.org
Il Pinguino - Teramo
Non disponibile
MarsicaLUG - Marsica
www.marsicalug.it
OpenLUG - L'Aquila
Non disponibile
Pescara LUG
www.pescaralug.org
Pineto LUG
www.pinetolug.org
Pollinux LUG - Pollutri
Non disponibile
SSVLUG - San Salvo, Vasto, Termoli
www.ssvlug.org
SulmonaLUG
http://sulmonalug.it
TeateLUG - Chieti
Non disponibile
TeLUG - Teramo
www.telug.it
User Group Valle Roveto
http://linuxvalley-os4.blogspot.com/

BASILICATA

Basilicata LUG - Potenza e Matera
www.baslug.org

CALABRIA

3BYLug - Trebisacce
www.3bylug.tk
Bogomips - Bisignano
www.blug.it
CastroLUG
http://castrolug.altervista.org
Cosenza Hack Laboratory
http://hacklab.cosenzainrete.it/
CSLUG - Cosenza
http://cslug.linux.it
CzLug
Non disponibile
HackLab Catanzaro
http://hacklab.cz
Piana LUG - Piana di Gioia Tauro
Non disponibile
Reggio Calabria LUG
http://rclug.linux.it
Revolutionary Mind
www.revolutionarymind.org
SpixLug - Spezzano Albanese
Non disponibile

CAMPANIA

AFR@Linux LUG
www.afralinux.netsons.org
Afralug - Afragola
www.afralug.com
CasertaLUG
www.casertaglug.org
Hackaserta 81100
www.81100.eu.org
HackMeetNaples Napoli HackLab
www1.autistici.org/hmn
IGLUG - Napoli e provincia
www.iglug.org
IRLUG - Irpinia
www.irlug.it
LUG-Ischia
www.lug-ischia.org
NALUG - Napoli
www.nalug.net
Neapolis Hacklab
www.officina99.org/hacklab.html
Padulug - Paduli (BN)
http://linux.paduli.com
SCALUG - Scafati (SA)
http://xoomer.alice.it/scalug/
Tuxway.org - Provincia di Napoli
www.tuxway.org
VaLug - Vallo Linux User Group
www.valug.it
XALUG - Salerno
http://xalug.tuxlab.org

EMILIA ROMAGNA

ALFLUG - Alfonsine
www.alflug.it
Borgotaro LUG - Val Taro
http://btlug.it/
ConoscereLinux - Modena
www.conoscerelinux.it
ERLUG
http://erlug.linux.it
Ferrara LUG
www.ferrara.linux.it
FoLUG - Forlì
http://folug.linux.it
ImoLUG - Imola
www.imolug.org
LUGPiacenza
www.lugpiacenza.org
PANLUG - Vignola
Non disponibile
PLUG - Parma
http://parma.linux.it
RavennaLUG
www.ravennalug.org
RELug - Reggio Emilia e provincia
http://relug.linux.it
RiminiLug
www.riminilug.it
S.P.R.I.Te
http://sprite.csr.unibo.it
UIELinux - Valle del Rubicone
www.uielinux.org

FRIULI VENEZIA GIULIA

CarniaLUG - Tolmezzo
www.carnialug.net
GOLUG - Gorizia
www.golug.it
IGLU - Udine
http://iglu.cc.uniud.it
LUG Pordenone
www.pnlug.it
LugTrieste
http://trieste.linux.it
LUG [A] [L] [P] - Aquileia
www.alproject.org

LAZIO

CiLUG - Frosinone
www.cilug.org
CLUG - Cassino
http://cassino.linux.it/
GioveLUG - Terracina
www.giovelug.org
La Sapienza LUG
www.lslug.org
Latina LUG
www.llg.it
LUG Privernum Volsca - Priverno (LT)
www.pvlug.org
LUGRoma
www.lugroma.org
LUGRoma 3
www.lugroma3.org
TorLUG - Università Tor Vergata - Roma
http://lug.uniroma2.it/
V.I.S.C.O.S.A. - Ciampino
www.viscosa.org

LIGURIA

Genuense Lug - Genova e d'intorni
http://genova.linux.it
GinLug - Genova Sampierdarena
www.sennaweb.org
Govonis GNU/LUG - Provincia di Savona
www.govonis.org
SLIMP - Software Libero Imperia
http://slimp.it/
TLug-TSL - Tigullio Ligure
http://tlug.linux.it/

LOMBARDIA

BGLug - Bergamo e provincia
www.bglug.it
BGLug Valle Seriana - Valle Seriana
http://bglugvs.web3king.com/
BrigX - Monza e Brianza
http://brigx.it
GL-Como - Como
www.gl-como.it
GLUX - Lecco e provincia
www.lecco.linux.it
GULLP - Gruppo Utenti Linux Lonate Pozzolo
www.gullp.it
IspraLUG - Ispra
http://ispralug.eu/
LIFO - Varese
www.lifolab.org
LIFOS - Cinisello Balsamo
www.lifos.org
Linux Var - Varese
www.linuxvar.it
LoLug - Lodi e provincia
www.lolug.org
Lug Bocconi - Milano
www.lug-bocconi.org
LugBS - Brescia e provincia
http://lugbs.linux.it/
Lug Castegnato - Castegnato

L'eco dei LUG

www.kenparker.eu/LugCastegnato
LugCR - Cremona e provincia
www.lugcr.it
Lug Crema - Crema
http://filibusta.crema.unimi.it/
LUGDucale - Vigevano
www.lugducale.it
LugMan - Mantova e provincia
www.lugman.org
LugOB - Cologne e ovest bresciano
www.lugob.org
MoBLUG - Monza e Brianza
www.bubblesfactory.it
OpenLabs - Milano
www.openlabs.it
POuL - Milano
www.poul.org
TiLug - Pavia
http://pavia.linux.it
VigLug - Vignate, Milano Est - Adda Martesana
www.viglug.org

MARCHE
Ascolinux LUG/FSUG Ascoli
http://marche.linux.it/ascoli/
CameLUG - Camerino
www.camelug.it
CMlug
www.cmlug.org
Egloo
www.egloo.org
FanoLUG
www.fanolug.org
Fermo LUG
www.linuxfm.org/fermolug/
GLM - Macerata
www.gruppolinuxmc.it/start/index.php
LUG Ancona
www.egloo.org
LUG Marche
http://marche.linux.it
PDP Free Software User Group
http://pdp.linux.it
SenaLug - Senigallia
www.lug.senigallia.biz

MOLISE
Campobasso LUG
http://cb.linux.it/
FrenterLUG - Larino
Non disponibile
SmaLUG - San Martino
www.smalug.org

PIEMONTE
ABC Lug - Alba/Bra/Carmagnola
http://abc.linux.it/
AlLug - Alessandria e provincia
www.allug.it
BiLUG - Provincia di Biella
http://www.bilug.it
FASoLi - Alessandria e provincia
http://softwarelibero.al.it/
Gallug - Galliate
www.gallug.it
GlugTO - Torino e provincia
www.torino.linux.it
IvLug - Ivrea Linux User Group
www.ivlug.it
Linox Novara
www.linoxnovara.org
SLIP - Pinerolo
http://pinerolo.linux.it/
ValSusinux - Val Susa e Val Sangone
www.valsusinux.it

PUGLIA
BriLUG - Brindisi
www.brilug.it
CapitanLUG - Capitanata
www.capitanlug.it
LATLUG - Latiano Linux User Group
www.latlug.org
LUGargano
www.lugargano.it
LUGBari - Bari e provincia
www.lugbari.org
MurgiaLug - Santeramo in Colle
www.open-pc.eu/index.php/murgialug/
SaLUG! - Salento
http://salug.it
Talug - Taranto
www.talug.it

SARDEGNA
GNUraghe - Oristano
www.gnuraghe.org
GULCh - Cagliari
www.gulch.it
PLUGS - Sassari
www.plugs.it
GULMh - Macomer (NU)
www.gulmh.org

SICILIA
CefaLug - Cefalù
http://cefalug.linux.it
cLUG - Caltanissetta
www.clug.it
EnnaLUG
www.ennalug.org
FreakNet MediaLab - Catania
www.freaknet.org
Leonforte LUG
http://leonforte.linux.it
LUG Catania
www.catania.linux.it
LUGSR - Siracusa
www.siracusa.linux.it
MELUG - Messina
Non disponibile
Norp LUG - Noto, Pachino, Rosolini
Non disponibile
PALUG - Palermo
http://palermo.linux.it
RgLUG - Ragusa e provincia
http://ragusa.linux.it
VPLUG Linux Planet - Provincia Caltanisetta
www.vplug.it
SputniX - Palermo
www.sputnix.it

TOSCANA
ACROS - Versilia, Lucca, Massa Carrara
www.lug-acros.org
Elbalinux
Non disponibile
ElsaGLUG - Val d'Elsa
www.elsaglug.org
FLUG - Firenze
www.firenze.linux.it
GOLEM - Empoli, Valdelsa
http://golem.linux.it
GroLUG - Grosseto
www.grolug.org
G.U.L.LI - Livorno
www.livorno.linux.it
GulP! Piombino
http://gulp.perlmonk.org
GULP Pisa
www.gulp.linux.it
GuruAtWork - Grosseto e provincia
www.guruatwork.com
IPIOS - Bibbiena e valle del Casentino
www.ipios.org
Lucca LUG
http://luccalug.it
L.U.G.A.R - Arezzo
Non disponibile
PLUG - Prato e provincia
www.prato.linux.it
PtLug - Pistoia e provincia
www.ptlug.org
SLUG - Siena e provincia
www.siena.linux.it

TRENTINO ALTO ADIGE
LinuxTrent - Trento
http://linuxtrent.it
LugBz - Bolzano
www.lugbz.org

UMBRIA
OrvietoLUG
www.orvietolug.it
LUG Perugia
www.perugiagnulug.org
TerniLUG
www.ternignulug.org

VALLE D'AOSTA
SLAG - Aosta
www.slag.it

VENETO
0421ug - Provincia di Venezia
www.0421ug.org
BLUG - Belluno
http://belluno.linux.it
Faber Libertatis - Padova
http://faberlibertatis.org
GrappaLUG - Bassano del Grappa
http://grappalug.homelinux.net/
ILC - Informatica Libera Cittadellese - FSUG
http://ilc.pd.it
LegnagoLUG
Non disponibile
Linux Ludus - Villafranca (VR)
www.linuxludus.it
LugAnegA
www.luganega.org
LUGSF - San Fidenzio
Non disponibile
LUG Vicenza
www.vicenza.linux.it
LugVR - Verona
www.verona.linux.it
MontelLUG - Montebelluna
www.montellug.it
FSUG Padova
www.fsugpadova.org
RoLUG - Rovigo
http://rovigo.linux.it
TVLUG - Treviso
www.tvlug.it
VELug - Venezia
www.velug.it
AViLUG Schio
http://www.avilug.it/doku.php

NAZIONALI
FSUGitalia
www.fsugitalia.org
Gentoo Channel Italia
www.gechi.it
MajaGLUG
www.majaglug.net
SkyLUG
http://tech.groups.yahoo.com/group/skylug/

In edicola l'11 dicembre

NEL PROSSIMO NUMERO

Hai un argomento da proporci? Scrivi a prossimamente@linux.it

RITORNO A UBUNTU

La distribuzione preferita fra i neofiti non è sempre quella che scelgono gli appassionati di Linux: è il momento di dare una chance a questa distro...

LINUX PRO

Mensile, prezzo di copertina 5,90 €
www.linuxpro.it

Direttore responsabile: *Luca Sprea*

Traduzione e Localizzazionea cura di:
Ventidodici di Andrea Orchesi
redazione@linuxpro.it


Sprea S.p.A.
Socio Unico - direzione e coordinamento di Sprea Holding S.p.A.

Presidente: Luca Sprea
Consigliere delegato: Mario Sprea

Coordinamento:
Gabriella Re (Foreign Rights) international@sprea.it, Alberta Rivolta (PA), Ambra Palermi (Segreteria Editoriale), Francesca Sigismondi (Ufficio Legale), Tiziana Rosato (acquisti e produzione), Emanuela Mapelli (Pianificazione Pubblicitaria)

Amministrazione: Anna Nese (CFO), Erika Colombo (controller), Irene Citino, Sara Palestra
amministrazione@sprea.it

Servizio qualità edicolanti e DL: Sonia Lancellotti, Andrea Palermi
distribuzione@sprea.it

PUBBLICITA' E MARKETING:
Alessandra Cappellacci- alessandracappellacci@sprea.it
Tel.0292432275 - Cell.3204670523

Sede Legale: - via Torino, 51 20063 Cernusco Sul Naviglio (Mi) - Italia
PI 12770820152- Iscrizione camera Commercio 00746350149

ABBONAMENTI E ARRETRATI

Servizio qualità abbonamenti e arretrati: Desirée Conti

Abbonamenti: si sottoscrivono on-line su www.linuxpro.it/abbonamenti
Mail: abbonamenti@linuxpro.it
Fax: 02 700 53 76 72
Tel: 02 87 15 82 25 (lun-ven / 9:00-18:00)

Arretrati: si acquistano on-line su www.linuxpro.it/arretrati
Mail: arretrati@linuxpro.it
Fax: 02 700 53 76 72 - Tel: 02 87 15 82 25 (lun-ven / 9:00-18:00)
Il prezzo dell'abbonamento è calcolato in modo etico perché sia un servizio utile e non in concorrenza con la distribuzione in edicola.

www.myabb.it a cura di Aktia srl
Contenuti su licenza: "Linux Format"- Future Publishing Limited PLC., Bath UK

Registrazione testata: Linux Pro, pubblicazione mensile registrata al Tribunale di Milano il 08.02.2003 con il numero 74.

Distributore per l'Italia e per l'estero:
Press-Di Distribuzione stampa e multimedia s.r.l. - 20134 Milano
ISSN: 1722-6163

Stampa: Arti Grafiche Boccia S.p.A.- Salerno